BAB II
TINJAUAN PUSTAKA

2.1 Profil Tempat Kerja Praktek

Profil tempat kerja praktek ini akan membahas tentang sejarah instansi dari PT.Medikom, badan hukum instansi dan struktur organisasi dan job description dari struktur organisasi yang ada di PT.Medikom terebut.

2.1.1 Sejarah Instansi

Surat Kabar Mingguan Medikom merupakan salah satu dari sekian banyaknya Koran lokal yang terbit di kota Bandung. Koran Mingguan ini didirikan pada tanggal 4 November 2002 di Jalan Peta (Lingkar Selatan) No. 149 A, Bandung. Kini Medikom bermarkas di Jalan Pratista Utara II No. 12 Antapani Bandung.

Mingguan Medikom didirikan oleh Yahya Siregar, Edie Nirwan, Tedie Subarsyah dan Kgs. Dedi Iskandar. Minguan Medikom saat ini dipimpin oleh Yahya Siregar sekaligus merangkap sebagai pimpinan redaksi.

Cakupan peredaran Mingguan Medikom dikhususkan untuk wilayah Kabupaten dan Kota Bandung, seluruh Kabupaten/Kota di Jawa barat, DKI Jakarta, Jawa tengah, Jawa Timur, Banten, Sumatera Utara, dan kepulauan Riau. Mingguan Medikom terbit setiap satu minggu sekali, tiap senin.

Kehadiran Medikom, sebagaimana surat kabar lainnya, secara praktis untuk menyuguhkan informasi khususnya berita lokal atau kedaerahan pada khalayak. Sebagai Koran lokal, Medikom berusaha mendekatkan diri dengan pembacanya dengan cara memuat berita atau informasi yang mereka butuhkan. Lingkungan terdekat, seperti kota bandung, merupakan yang pertama dirangkul karena di kota inilah Medikom berpijak.

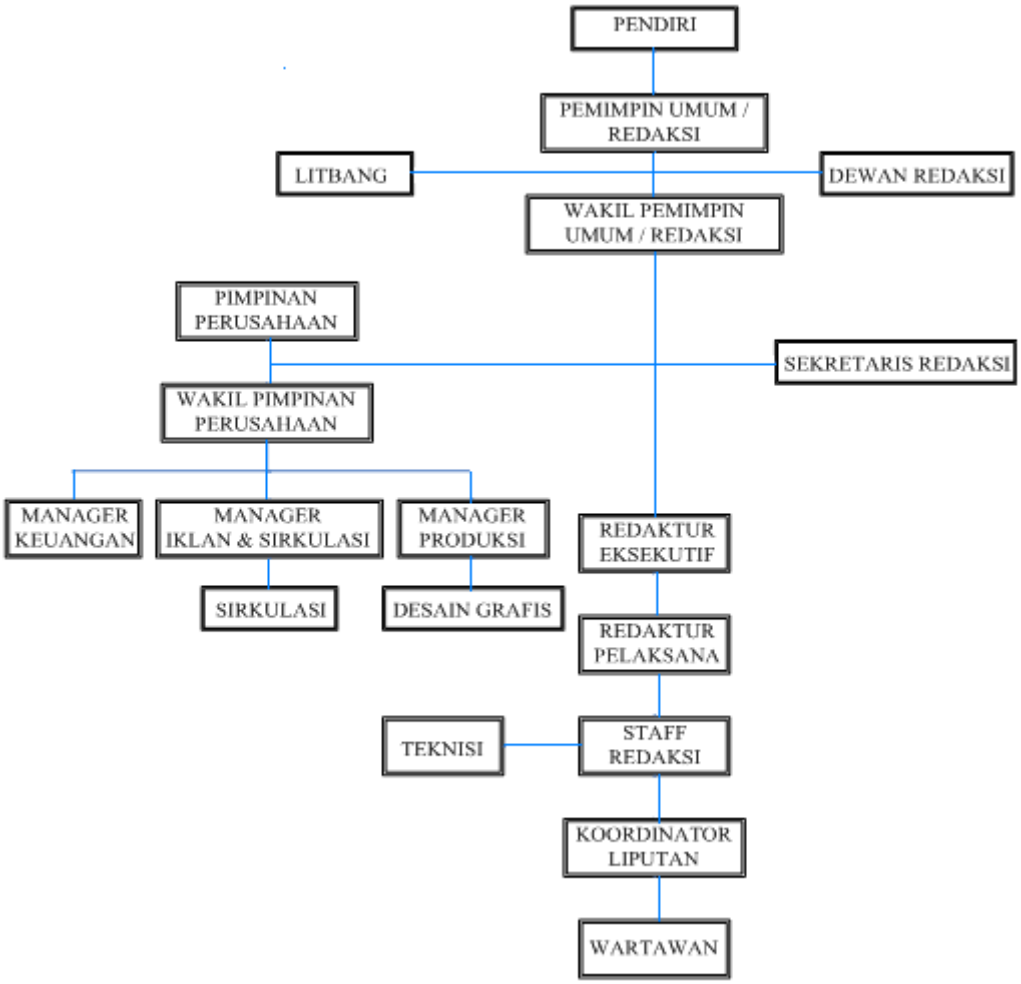
Gambar 2.1 Logo Perusahaan

Huruf i berwarna merah menandakan independen. Tepat dan perlu adalah Koran Medikom memang perlu untuk pembaca dan tepat untuk dibaca.

2.1.2 Badan Hukum Instansi

Diterbitkan: CV WIYATA PURNAKARYA. Berdasarkan UU No. 40/1999. Terdaftar pada Direktorat Merk Ditjen Hak dan Kekayaan Intelektual Departemen Kehakiman dan HAM RI No. 556756/22 Desember 2003.

2.1.3 Struktur Organisasi dan Job Description



Gambar 2.2 Struktur Organisasi

Job Description

a. Pemimpin Umum (*General Manager*)

Ia bertanggung jawab atas keseluruhan jalannya penerbitan pers, baik ke dalam maupun ke luar. Ia dapat melimpahkan pertanggungjawabannya terhadap hukum kepada Pemimpin Redaksi sepanjang menyangkut isi penerbitan (*redaksional*) dan kepada Pemimpin Usaha sepanjang menyangkut pengusahaan penerbitan.

b. Wakil Pemimpin Redaksi

Wakil Pemimpin Redaksi (*Editor in Chief*) bertanggung jawab terhadap mekanisme dan aktivitas kerja keredaksian sehari-hari. Ia harus mengawasi isi seluruh rubrik media massa yang dipimpinnya. Di surat kabar mana pun, Pemimpin Redaksi menetapkan kebijakan dan mengawasi seluruh kegiatan redaksional. Ia bertindak sebagai jenderal atau komandan yang perintah atau kebijakannya harus dipatuhi bawahannya. Kewenangan itu dimiliki katena ia harus bertanggung jawab jika pemberitaan medianya digugat pihak lain.

c. Dewan Redaksi

Dewan Redaksi biasanya beranggotakan Pemimpin Umum, Pemimpin Redaksi dan Wakilnya, Redaktur Pelaksana, dan orang-orang yang dipandang kompeten menjadi penasihat bagian redaksi. Dewan Redaksi bertugas memberi masukan kepada jajaran redaksi dalam melaksanakan pekerjaan redaksional. Dewan Redaksi pula yang mengatasi permasalahan penting redaksional, misalnya menyangkut berita yang sangat sensitif atau sesuai-tidaknya berita yang dibuat tersebut dengan visi dan misi penerbitan yang sudah disepakati.

d. Redaktur Eksekutif

Di bawah Pemred biasanya ada Redaktur Pelaksana (*Managing Editor*). Tanggung jawabnya hampir sama dengan Pemred/Wapemred, namun lebih bersifat teknis. Dialah yang memimpin langsung aktivitas peliputan dan pembuatan berita oleh para reporter dan editor.

e. Redaktur Pelaksana

Redaktur pelaksana (*editor*) sebuah penerbitan pers biasanya lebih dari satu. Tugas utamanya adalah melakukan editing atau penyuntingan, yakni 13 aktivitas penyeleksian dan perbaikan naskah yang akan dimuat atau disiarkan. Di internal redaksi, mereka disebut Redaktur Desk (*Desk Editor*), Redaktur Bidang, atau Redaktur Halaman karena bertanggung jawab penuh atas isi rubrik tertentu dan editingnya. Seorang redaktur biasanya menangani satu rubrik, misalnya rubrik ekonomi, luar negeri, olahraga, dsb. Karena itu ia dikenal pula dengan sebutan Jabrik atau penanggung jawab rubrik.

f. Staf Redaksi

Setingkat dengan Redaktur/Editor adalah Staf redaksi . Ia bertanggung jawab menangani Naskah Siap Cetak (All In Hand/All Up) dari para redaktur, yaitu semua naskah berita yang sudah diturunkan ke percetakan dan sudah diset bersih, desain cover dan perwajahan (tataletak, layout, artistik), dan hal-ihwal sebelum koran dicetak. Bagian lain di yang berada di bawah koordinasi Staf Redaksi adalah Setter atau juruketik naskah. Ia bertugas mengetik naskah yang akan dimuat. Ada pula Korektor yang bertugas mengoreksi (membetulkan) kesalahan ketik pada naskah yang siap cetak.

g. Wartawan

Di bawah para editor adalah para Wartawan. Mereka merupakan prajurit di bagian redaksi. Mencari berita lalu membuat atau menyusunnya, merupakan tugas pokoknya. Fotografer (wartawan foto atau jurupotret) tugasnya mengambil gambar peristiwa atau objek tertentu yang bernilai berita atau untuk melengkapi tulisan berita yang dibuat wartawan tulis. Ia merupakan mitra kerja yang setaraf dengan wartawan tulisan (reporter). Jika tugas wartawan tulis menghasilkan karya jurnalistik berupa tulisan berita, opini, atau feature, maka fotografer menghasilkan Foto Jurnalistik (Journalistic Photography, Photographic Communications). Fotografer menyampaikan informasi atau pesan melalui gambar yang ia potret. Fungsi foto

jurnalistik antara lain menginformasikan (to inform), meyakinkan (to persuade), dan menghibur (to entertain).

h. Litbang

Bagian yang tak kalah pentingnya untuk membantu kelancaran kerja redaksi adalah bagian Perpustakaan dan Dokumentasi serta bagian Penelitian dan Pengembangan (Litbang). Litbang memantau perkembangan sebuah penerbitan, survei pembaca, dan memberikan masukan-masukan bagi pengembangan redaksional dan bagian lainnya, termasuk pembinaan dan pengembangan kualitas sumber daya manusia.

i. Manager Iklan dan Sirkulasi

Bertugas menyebarluaskan media massa, yakni melakukan pemasaran (marketing) atau penjualan (saling) media massa. Bagian ini merupakan sisi komersial meliputi sirkulasi/distribusi, iklan, dan promosi. Biasanya, bagian pemasaran dipimpin oleh seorang Pemimpin Perusahaan atau seorang Manajer Pemasaran (Marketing Manager) yang membawahkan Manajer Sirkulasi, Manajer Iklan, dan Manajer Promosi.

j. Manager Produksi

Manager bertanggung jawab untuk memfasilitasi, mengatur, mengontrol dan meningkatkan kemampuan sumberdaya manusia ,bahan baku setengah jadi/jadi dan mesin – mesin produksi didalam wilayah tanggung jawabnya guna memaksimalkan efisiensi, meminimalkan biaya dan menghasilkan bahan setengah jadi / jadi yang memenuhi standard kebutuhan pelanggan.

2.2 Landasan Teori

Landasan teori ini berisikan teori mengenai sistem informasi, basis data, jaringan komputer, internet, metode pengembangan perangkat lunak, html, java script dan bahasa pemrograman PHP.

2.2.1 Sistem Informasi

Dalam pengertian sistem informasi ada baiknya terlebih dahulu mengerti dari sistem dan informasi. Dengan memahami defenisi sistem dan informasi akan dapat diperoleh penjelasan yang baik tentang sistem informasi.

Pengertian Sistem

Istilah sistem merupakan suatu istilah yang tidak asing lagi dan banyak digunakan secara luas pada lembaga-lembaga atau bidang-bidang ilmu pengetahuan dan teknologi. Istilah sistem berasal dari bahasa Yunani yaitu *Sistema*. Ditinjau dari asal katanya, sistem berarati sekumpulan objek yang bekerja bersama-sama untuk menghasilkan suatu kesatuan metode yang digabungkan dan diatur sedemikian rupa yang berfungsi mencapai tujuan. Suatu sistem dapat terdiri dari beberapa subsistem yang saling berhubungan membentuk suatu kesatuan sehingga sasaran atau tujuan sistem tersebut dapat tercapai. Pada umumnya ciri-ciri sistem bertujuan, punya batas, terbuka, tersusun dari subsistem, dan saling berkaitan dan saling ketergantungan, merupakan suatu kebulatan yang utuh, melakukan kegiatan transformasi, ada mekanisme kontrol, dan memiliki kemampuan mengatur dan menyesuaikan diri sendiri.

Pengertian Informasi

Informasi merupakan data yang telah diproses, transformasi data, sehingga berubah bentuk menjadi informasi. Informasi adalah data yang telah diambil kembali, diolah, atau sebaliknya digunakan untuk tujuan *informative*, argumentasi, ataupun sebagai dasar untuk pengambilan keputusan. Informasi ini adalah hasil proses data yang bentuknya kurang berguna menjadi data yang berguna. Menurut Jogiyanto Hartono (1999, hal :696) kualitas suatu informasi tergantung dari tiga hal, yaitu informasi harus akurat, tepat pada waktunya dan relevan.

1. Akurat berarti informasi harus bebas dari kesalahan-kesalahan dan tidak menyesatkan.
2. Tepat pada waktunya berarti informasi yang datang pada penerima tidak boleh terlambat.

3. Relevan berarti informasi tersebut mempunyai manfaat untuk pemakainya.

Kegunaan informasi itu sendiri adalah untuk memberitahukan ke pengguna informasi mengenai suatu masalah agar pengguna informasi lebih dapat menguasai masalah yang dihadapinya. Informasi juga ketidakpastian tentang suatu masalah yang dapat digunakan untuk memilih resiko yang paling kecil dan keuntungan yang besar dalam pemilihan alternatif bagi suatu proses pengambilan keputusan.

Pengertian Sistem Informasi

Dari definisi sistem dan informasi dapat disimpulkan bahwa sistem informasi adalah suatu sistem di dalam suatu sistem organisasi yang merupakan kombinasi dari orang-orang, fasilitas, teknologi, media, prosedur-prosedur, dan pengendalian, yang ditujukan untuk mendapatkan jalur komunikasi penting, memproses tipe transaksi rutin tertentu, memberi sinyal kepada manajemen dan yang lainnya terhadap kejadian-kejadian internal dan eksternal yang penting dan menyediakan sutau dasar informasi untuk pengambilan keputusan yang cerdik (Jogiyanto Hartono, 1999).

Peranan Komputer dalam Sistem Informasi

Komputer mempunyai peranan penting untuk membantu manusia dalam menyelesaikan dan memecahkan masalah yang dihadapinya. Setiap pemecahan persoalan yang dilakukan mansia, hendaknya mempunyai dua unsur, yaitu :

1. Adanya fakta dan data.
2. Bagaimana proses pemecahannya.

Jika pengelohan data dilakukan secara manual tidak akan menjamin ketelitian serta kebenaran hasil atau informasi yang diinginkan. Masalah pendataan dan pengelohan data dapat dipecahkan dengan mudah dalam waktu singkat dengan menggunakan alat bantu komputer.

Dalam hal inilah perlu diperhatikan suatu sistem pengolahan data yang cepat dan tepat untuk memberikan informasi yang sangat menjamin benarnya keputusan yang telah ditetapkan. Dilihat dari aspek praktisnya, maka tidak jarang seseorang terlambat atau belum selesai didalam mengambil keputusan dan mendapat informasi

yang cepat dalam pengolahan komputer. Peranan komputer dalam suatu sistem informasi sangatlah penting. Hal ini diakibatkan besar dan banyaknya data yang akan diolah dengan beraneka ragam jenis data. Disinilah komputer memegang peranan penting untuk melakukan pengolahan data yang banyak dan beraneka ragam informasi dapat diperoleh dengan cepat dan tepat.

2.2.2 Basis Data

Data, Informasi dan Basis Data

Data merupakan fakta mengenai suatu objek seperti manusia, benda, peristiwa, konsep, keadaan dan sebagainya yang dapat dicatat dan mempunyai arti secara implisit. Data dapat dinyatakan dalam bentuk angka, karakter atau simbol, sehingga bila data dikumpulkan dan saling berhubungan maka dikenal dengan istilah basis data (database) [Ramez2000]. Sedangkan menurut George Tsu-der Chou basis data merupakan kumpulan informasi bermanfaat yang diorganisasikan ke dalam aturan yang khusus. Informasi ini adalah data yang telah diorganisasikan ke dalam bentuk yang sesuai dengan kebutuhan seseorang [Abdul1999]. Menurut *Encyclopedia of Computer Science and Engineer*, para ilmuwan di bidang informasi menerima definisi standar informasi yaitu data yang digunakan dalam pengambilan keputusan.

Definisi lain dari basis data menurut Fabbri dan Schwab adalah sistem berkas terpadu yang dirancang terutama untuk meminimalkan duplikasi data.

Menurut Ramez Elmasri mendefinisikan basis data lebih dibatasi pada arti implisit yang khusus, yaitu:

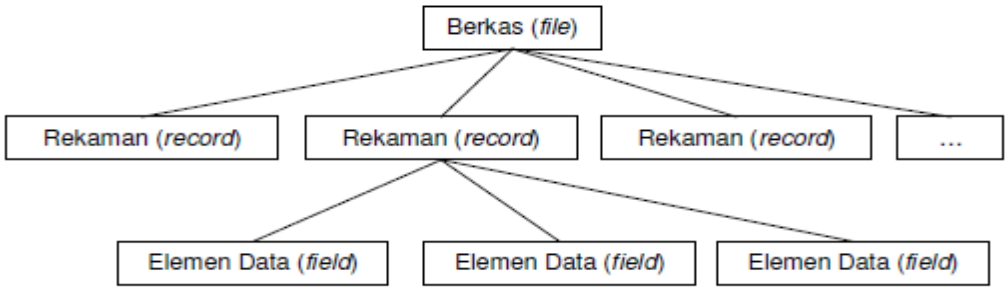
a. Basis data merupakan penyajian suatu aspek dari dunia nyata (*real world*).
b. Basis data merupakan kumpulan data dari berbagai sumber yang secara logika mempunyai arti implisit. Sehingga data yang terkumpul secara acak dan tanpa mempunyai arti, tidak dapat disebut basis data.
c. Basis data perlu dirancang, dibangun dan data dikumpulkan untuk suatu tujuan. Basis data dapat digunakan oleh beberapa *user* dan beberapa aplikasi yang sesuai dengan kepentingan *user*.

Dari beberapa definisi-definisi tersebut, dapat dikatakan bahwa basis data memounyai berbagai sumber data dalam pengumpulan data, bervariasi derajat interaksi kejadian dari dunia nyata, dirancang dan dibangun agar dapat digunakan oleh beberapa user untuk berbagai kepentingan [Waliyanto2000].

Hirarki Data

Data diorganisasikan kedalam bentuk elemen data (*field*), rekaman (*record*), dan berkas (*file*). Definisi dari ketiganya adalah sebagai berikut:

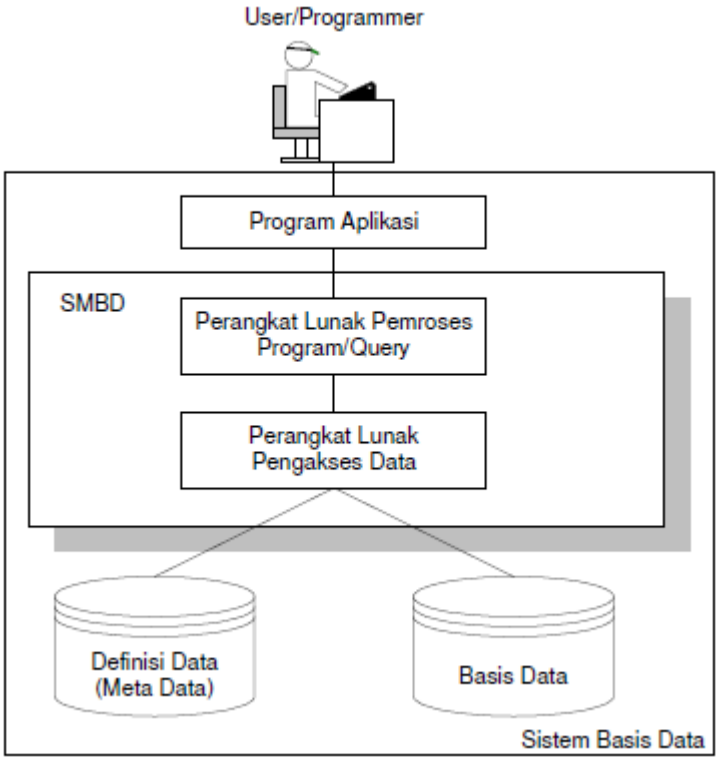
- Elemen data adalah satuan data terkecil yang tidak dapat dipecah lagi menjadi unit lain yang bermakna. Misalnya data siswa terdiri dari NIS, Nama, Alamat, Telepon atau Jenis Kelamin.
- Rekaman merupakan gabungan sejumlah elemen data yang saling terkait. Istilah lain dari rekaman adalah baris atau tupel.
- Berkas adalah himpunan seluruh rekaman yang bertipe sama.



Gambar 2.3 Hirarki Data

Sistem Basis Data

Gabungan antara basis data dan perangkat lunak SMBD (Sistem Manajemen Basis Data) termasuk di dalamnya program aplikasi yang dibuat dan bekerja dalam satu sistem disebut dengan Sistem Basis Data.



Gambar 2.4 Konsep Sistem Basis Data

C. J. Date menyatakan bahwa sistem basis data dapat dianggap sebagai tempat untuk sekumpulan berkas data yang terkomputerisasi dengan tujuan untuk memelihara informasi dan membuat informasi tersebut tersedia saat dibutuhkan.

Data Base Management System (DBMS)/Sistem Manajemen Basis Data (SMB)

DBMS dapat diartikan sebagai program komputer yang digunakan untuk memasukkan, mengubah, menghapus, memodifikasi dan memperoleh data/informasi dengan praktis dan efisien.

Kelebihan dari DBMS antara lain adalah:

- **Kepraktisan**. DBMS menyediakan media penyimpan permanen yang berukuran kecil namun banyak menyimpan data jika dibandingkan dengan menggunakan kertas.
- **Kecepatan**. Komputer dapat mencari dan menampilkan informasi yang dibutuhkan dengan cepat.

- **Mengurangi kejemuan**. Pekerjaan yang berulang-ulang dapat menimbulkan kebosanan bagi manusia, sedangkan mesin tidak merasakannya.
- **Update to date**. Informasi yang tersedia selalu berubah dan akurat setiap.

Keuntungan-keuntungan dalam penggunaan DBMS antara lain adalah:

a. Pemusatan kontrol data. Dengan satu DBMS di bawah kontrol satu orang atau kelkompok dapat menjamin terpeliharanya standar kualitas data dan keamanan batas penggunaannya serta dapat menetralkan konflik yang terjadi dalam persyaratan data dan integritas data dapat terjaga.
b. Pemakaian data bersama (Shared Data). Informasi yang ada dalam basis data dapat digunakan lebih efektif dengan pemakaian beberapa user dengan kontrol data yang terjaga.
c. Data yang bebas (independent). Program aplikasi terpisah dengan data yang disimpan dalam komputer.
d. Kemudahan dalam pembuatan program aplikasi baru.
e. Pemakaian secara langsung. DBMS menyediakan interface yang memudahkan pengguna dalam mengolah data.
f. Data yang berlebihan dapat dikontrol. Data yang dimasukkan dapat terjadi kerangkapan (*redudant*), untuk itu DBMS berfungsi untuk menurunkan tingkat redudancy dan pengelolaan proses pembaruan data.
g. Pandangan user (*user view*). Ada kemungkinan basis data yang diakses adalah sama, maka DBMS mampu mengatur interface yang berbeda dan disesuaikan dengan pemahaman tiap user terhadap basis data menurut kebutuhan.

Kelemahan-kelemahan DBMS antara lain:

a. Biaya. Kebutuhan untuk medapatkan perangkat lunak dan perangkat keras yang tepat cukup mahal, termasuk biaya pemeliharaan dan sumber daya manusia yang mengelola basis data tersebut.
b. Sangat kompleks. Sistem basis data lebih kompleks dibandingkan dengan proses berkas, sehingga dapat mudah terjadinya kesalahan dan semakin sulit dalam pemeliharaan data.

c. Resiko data yang terpusat. Data yang terpusat dalam satu lokasi dapat beresiko kehilangan data selama proses aplikasi.

Model Data

Model data dapat dikelompokkan berdasarkan konsep pembuatan deskripsi struktur basis data, yaitu:

a. Model data konsepsual (*high level*) menyajikan konsep tentang bagaiman user memandang atau memperlakukan data. Dalam model ini dikenalkan tiga konsep penyajian data yaitu:

   • ***Entity*** (entitas) merupakan penyajian obyek, kejadian atau konsep dunia nyata yang keberadaannya secara eksplisit didefinisikan dan disimpan dalam basis data, contohnya Mahasiswa, Matakuliah, Dosen, Nilai dan lain sebagainya.

   • ***Atribute*** (atribut) adalah keterangan-keterangan yang menjelaskan karakteristik dari suatu entitas seperti NIM, Nama, Fakultas, Jurusan untuk entitas Mahasiswa.

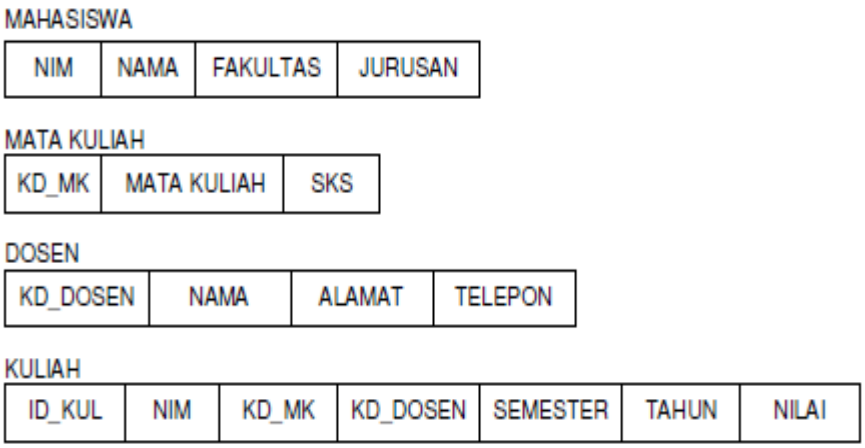
   • ***Relationship*** (hubungan) merupakan hubungan atau interaksi antara satu entitas dengan yang lainnya, misalnya entitas pelanggan berhubungan dengan entitas barang yang dibelinya.

b. Model data fiskal (*low level*) merupakan konsep bagaimana deskripsi detail data disimpan kedalam komputer dengan menyajikan informasi tentang format rekaman, urutan rekaman, dan jalur pengaksesan data yang dapat membuat pemcarian rekaman data lebih efisien.

c. Model data implementasi (*representational*) merupakan konsep deskripsi data disimpan dalam komputer dengan menyembunyikan sebagian detail deskripsi data sehingga para user mendapat gambaran global bagaimana data disimpan dalam komputer. Model ini merupakan konsep model data yang digunakan oleh model hirarki, jaringan dan relasional.

Skema dan Instan Basis Data

Skema basis data merupakan deskripsi dari basis data yang spesifikasinya ditentukan dalam tahap perancangan namun tidak terlalu diharapkan diubah setiap saat. Penggambaran skema umumnya hanya berisi sebagian dari detail deskripsi basis data.



Gambar 2.5 Skema Basis Data Akademi

Sekelompok data yang tersusun dalam satu baris rekaman (*record/tuple)* dan tersimpan dalam basis data disebut dengan instansi (*instance*) atau kejadian (*occurences*).

Arsitektur DBMS

Arsitektur ini dikenal dengan nama arsitektur tiga skema (*three-schema architecture*) dimana fungsi ini untuk memisahkan antara basis data fisik dengan program aplikasi user. Skema-skema tersebut adalah sebagai berikut:

a. Level internal merupakan skema internal yang memuat deskripsi struktur penyimpanan basis data dan menggunakan model data fisikal serta mendefinisikan secara detail penyimpanan data dalam basis data, serta jalur pengaksesan data.
b. Level konsepsual adalah skema yang memuat deskripsi struktur basis data secara keseluruhan untuk semua pemakai. Skema ini hanya memuat deskripsi tentang entitas, atribut, hubungan dan batasan, tanpa memuat deskripsi data secara detail.

c. Level eksternal merupakan skema eksternal (*user view*) yang mendefinisikan pandangan data terhadap sekelompok user (*local view*) dengan menyembunyikan data lain yang tidak diperlukan oleh kelompok user tersebut.

Keuntungan dari arsitektur ini antara lain:

a. Perubahan skema konsepsual, yaitu adanya perubahan dalam skema konsepsual contohnya penambahan suatu item data tidak akan berpengaruh pada program aplikasi. Tetapi jika skema eksternal tidak sesuai lagi dengan skema konsepsual yang baru maka program aplikasi harus disesuaikan juga.
b. Perubahan skema internal. Pemisahan antara skema eksternal dan skema internal berfungsi untuk menjaga bila terjadi perubahan skema internal, misalnya ada penambahan "pointer" pada rekaman tidak memerlukan perubahan pada aplikasi.
c. Perubahan skema eksternal. Adanya penambahan skema eksternal atau pembuatan skema eksternal baru tidak akan berpengaruh pada aplikasi yang ada selama aplikasi tersebut tidak mengakses data berdasarkan skema yang baru.

Komponen DBMS

Komponen-komponen DBMS (Howe,1991) terdiri dari:

- Interface, yang didalamnya terdapat bahasa manipulasi data (*data manipulation language*)
- Bahasa definisi data (*data definition language*) untuk skema eksternal, skema konsepsual dan skema internal.
- Sistem kontrol basis data (*Database Control System*) yang mengakses basis data karena adanya perintah dari bahasa manipulasi data.

Contoh bahasa menggunakan komponen-komponen tersebut adalah SQL (*Structured Query Language*). SQL merupakan bahasa standar yang digunakan oleh kebanykan aplikasi-aplikasi DBMS.

Klasifikasi DBMS

Sistem Basisi Data dapat diklasifikasikan menjadi tiga bagian, yang terdiri dari:

a. Klasifikasi berdasarkan model data. Klasifikasi ini terdiri dari model data hirarki, model data jaringan, model data relasional.

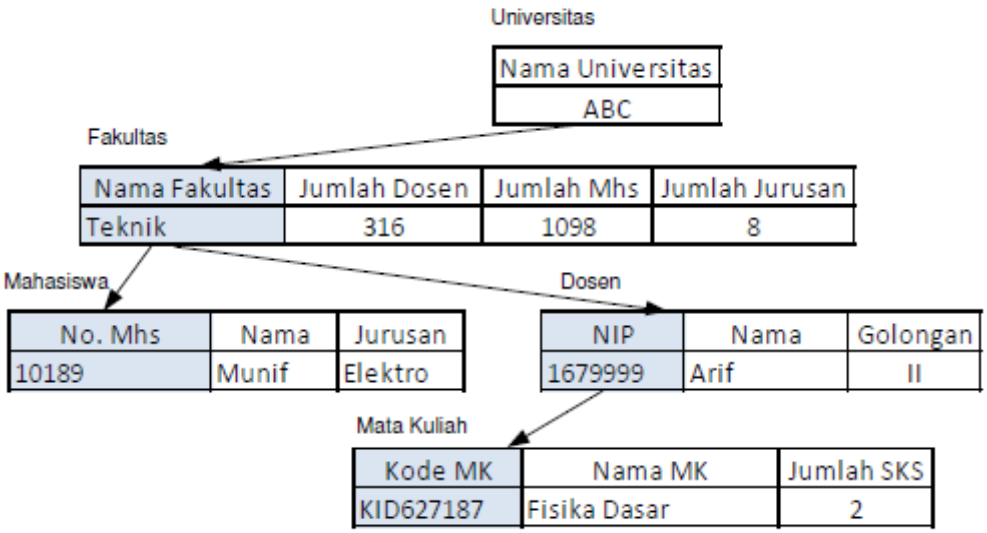
1. Model data hirarki

Dalam model ini, data disusun menurut struktur pohon yang merupakan bentuk lain dari abstraksi data untuk basis data akademi. Pada puncak hirarki diesbut dengan akar (*root*). Tiap entitas tingkat atas (*parent*) mempunyai satu atau lebih sub-entitas (*children*) sehingga setiap entitas hanya boleh mempunyai satu induk, tetapi dapat mempunyai banyak anak.

Pada mode data hirarki, hubungan antar entitas dinyatakan dalam satu-banyak (*one to many*) atau satu-satu (*one to one*). Dalam satu Universitas terdapat banyak Fakultas dan setiap Fakultas terdapat banyak Dosen atau banyak Mahasiswa, dan seterusnya. Tanda panah menunjukkan derajat keterhubungan "banyak".

Untuk menampilkan semua mata kuliah pada Fakultas tertentu harus dilakukan dalam dua tahap. Yang pertama adalah menampilkan rekaman semua Dosen yang mengajar di Fakultas tersebut, kemudian baru mata kuliah yang dipegang oleh para Dosen. Dalam hal ini penampilan data terlihat kurang efisien, sebab menggunakan entitas perantara (dosen) yang harus ditampilkan juga. Dikarenakan kunci data yang digunakan untuk menghubungkan antar entitas diberi kode dalam struktur data, maka untuk jumlah entitas perantara yang sedikit masih dapat dikatakan efisien.

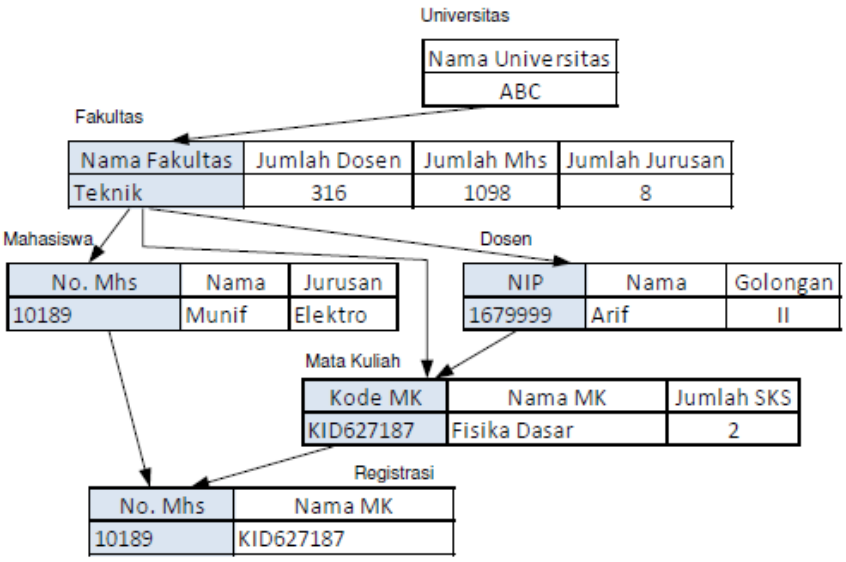
Kelemahan lain pada model data hirarki adalah tidak dapat melakukan pencarian data pada field. Misalnya dalam entitas mata ki\uliha tida pat ditampilkan hanya mata kuliah dengan jumlah SKS tertentu, sebab field "Jumlah SKS" bukan sebagai kunci data. Hal ini masih dapat dilakukan dengan mengubah struktur data dengan memberi hubungan khusus yang digunakan untuk mengubah struktur database. Kelebihan model ini adalah sangat mudah dipahami dan mudah dalam pembaharuan data.



Gambar 2.6 Organisasi rekaman data pada model hirarki

2. Model data Jaringan

Dalam model ini setiap entitas dapat mempunyai banyak induk dan banyak anak. Pada gambar menunjukkan entitas mata kuliah mempunyai dua induk, yaitu langsung berhubungan dengan Fakultas dan Dosen.



Gambar 2.7 Organisasi rekaman data model jaringan

Dalam model ini lebih sedikit terdapat data rangkap, namun lebih banyak terdapat hubungan antar entitas, sehingga akan menambah informasi hubungan yang harus disimpan dalam *database*. hal ini akan menambah volume dan kerumitan dalam penyimpanan berkas data.

3. Model data Relasional

Dalam model ini setiap field dapat dijadikan kunci data. Data rekaman disusun dari nilai yang berhubungan (*record*). Baris-baris ini akan membentuk tabel yang umunya tersimpan dalam satu berkas (*file*).

Tabel 2.8 Organisasi basis data model relasional

Mata Kuliah

No MK	Nama MK	SKS	NIP
KDI1892	Bahasa Inggris	2	1222
KAI6522	Fisika Lanjut	2	2344
...	...	...	...

Fakultas

Fakultas	Jml_Dsn	Jml_Mhs	Jml_Jur
Teknik	150	6320	3
Ekonomi	200	4120	2
...	...	...	...

Mahasiswa

No Mhs	Nama	Jurusan	Fakultas
11782	Arif	Elektro	Teknik
44366	Rudian	Mesin	Teknik
89878	Nita	Sipil	Teknik
...	...	...	...

Dosen

NIP	Nama	Gol
23666	Murdiansyah	III
132455	Johan	II
...	...	...

Registrasi

No MK	No Mhs
KDI1892	11782
KAI6522	44366
KAI6522	89878

Dengan menggunakan model ini, pencarian field dari suatu tabel atau banyak tabel dapat dilakukan dengan cepat. Pencarian atribut yang berhubungan pada tabel yang berbeda dapat dilakukan dengan menghubungkan terlebih dahulu tabel-tabel tersebut dengan menggunakan atribut yang sama (*joint operation*).

Keuntungan yang didapat dengan menggunakan model ini adalah sebagai berikut:

- Model ini lebih luwes karena nilai data dalam tabel tidak ada pembatasan dalam berbagai proses pencarian data.
- Model ini mempunyai latar belakang teori matematika.
- Pengorganisasian model relasional sangat sederhana, sehingga mudah dipahami.
- Basis data yang sama biasanya dapat disajikan dengan lebih sedikit terjadi data rangkap (*redudancy data*).

Sedangkan beberapa kelemahan model ini adalah:

- Lebih sulit dalam implementasinya terutama untuk data dengan jumlah yang besar dan tingkat kompleksitasnya tinggi.
- Proses pencarian informasi lebih lambat, karena beberapa tabel tidak dihubungkan secara fisik. Dalam manipulasi data yang menggunakan beberapa tabel akan memerlukan waktu yang lama, karena tabel-tabel harus dihubungkan terlebih dahulu.

b. Klasifkasi berdasarkan lokasi penyimpanan data, yaitu DBMS terpusat dan DBMS terdistribusi. Dalam DBMS terpusat basis data disimpan dalam satu komputer media penyimpan sehingga pengguuna sistem mengakses data dari pusat. DBMS terdistribusi, basis data tersebar pada penyimpanan tiap terminal pengguna (client). Antar pengguna dapat mengakses data secara langsung tanpa perlu melalui pusat penyimpanan. DBMS ini memerlukan sistem kontrol yang rumit.

c. Klasifikasi berdasarkan tujuan DBMS digunakan yaitu tujuan umum (*general purpose*) dan tujuan khusus (*special purpose*). Untuk tujuan umum dapat digunakan untuk berbagai tujuan dengan memperlakukan data sama menurut penggunaannya contoh aplikasinya adalah DBASE, ORACLE, FOXBASE dan sebagainya. DBMS tujuan khusus dirancang dan digunakan untuk keperluan tertentu, sebagai contoh pengelolaan data karyawan pada perusahaan Asuransi.

Pengembangan Database

Database diproses oleh DBMS untuk digunakan oleh pengembang maupun pengguna, yang mengakses DBMS secara langsung atau tidak langsung melalui program-program aplikasi. Database terdiri dari empat elemen utama yaitu data pengguna, metadata, indeks dan metadata aplikasi.

Data Pengguna

Hampir semua database me-representasikan data pengguna sebagai relasi dengan menganggapnya sebagai tabel data. Kolom dalam tabel berisi field-field atau

atribut dan baris tabel berisi record/tuple (rekaman) untuk keterangan entitas dalam lingkungan bisnis. Tidak semua relasi diperlukan, beberapa relasi lebih baik distrukturkan dengan proses normalisasi. Relasi ini dapat digambarkan dengan bentuk hubungan antara pelajar dengan guru sebagai berikut:

Tabel 2.9 Relasi Pelajar dengan Guru (R1)

NamaPelajar	TeleponPelajar	NamaGuru	TeleponGuru
Aminudin	7778889	Pardi	7789665
Usman	7896532	Pardi	7789665
Ari	7474856	Dadang	8965555
Rina	7895654	Marni	4562211
Tuti	7897744	Dadang	8965555
Joni	7845644	Dadang	8965555

Struktur relasi tersebut dapat terjadi beberapa masalah, misalnya jika guru Dadang mengganti nomor telepon maka tiga record yang terdapat guru Dadang diatas harus diganti juga. Untuk itu lebih baik jika struktur relasi diubah menjadi dua relasi seperti di bawah ini:

Tabel 2.10 Hubungan antara R1 dan R2

NamaPelajar	TeleponPelajar	NamaGuru
Aminudin	7778889	Pardi
Usman	7896532	Pardi
Ari	7474856	Dadang
Rina	7895654	Marni
Tuti	7897744	Dadang
Joni	7845644	Dadang

NamaGuru	TeleponGuru
Pardi	7789665
Dadang	8965555
Marni	4562211

Dari relasi diatas akan pengubahan data hanya dilakukan pada relasi kedua.

Metadata

Penjelasan struktur dari suatu tabel disebut dengan *metadata* dan terkadang disebut dengan *system tables*. Bentuk dari metada dapat digambarkan seperti

dibawah ini yang terdiri dua tabel. Tabel pertama berisi daftar tabel-tabel di dalam suatu database sedangkan tabel yang kedua berisi daftar kolom-kolom pada suatu tabel.

Tabel 2.11 Tabel SysTable

Nama Tabel	Jumlah Kolom	Primary Key
Pelajar	4	NIS
Guru	3	NIP
Mata Pelajaran	4	Kode_MP
Relasi Belajar	3	{NIS,Kode_MP,NIP}

Tabel 2.12 Tabel SysColumns

Nama Kolom	Nama Tabel	Tipe Data	Panjang
NIS	Pelajar	String	5
Nama	Pelajar	String	20
Telepon	Pelajar	String	12
Alamat	Pelajar	String	50
NIP	Guru	String	6
Nama	Guru	String	20
Telepon	Guru	String	12
Divisi	Guru	String	20
Kode_MP	Mata Pelajaran	String	5
Nama MP	Mata Pelajaran	String	15
Jumlah Jam	Mata Pelajaran	Integer	4
NIS	Relasi Belajar	String	5
Kode_MP	Relasi Belajar	String	5
NIP	Relasi Belajar	String	6
Tingkat	Relasi Belajar	String	2

Indeks

Tipe database ini digunakan untuk meningkatkan kinerja dan akses suatu database. Terkadang tipe data ini disebut dengan *overhead data*, terdiri dari prinsip-prinsip indeks serta beberapa penggunaan struktur data *link list*. Di bawah ini contoh pengguanan dua buah indeks dari tabel Mahasiswa:

Tabel 2.6 Contoh Tabel Mahasiswa

NO	Nama	Jurusan	Kelas
10	David Carradine	Akuntansi	2AB
20	Jaka Sembung	Manajemen	2CV
30	Kebo Ireng	Manajemen	2CV
40	Lasmini	Teknik Sipil	1SP
50	Joni Keboy	Akuntansi	1AB
60	Franc De Nero	Manajemen	2AB
70	Marco Van Basten	Teknik Sipil	1SP
80	Maradani	Teknik Sipil	1SP
90	Dona Doni	Akuntansi	1AB

Tabel 2.7 Tabel Indeks berdasarkan Kelas

Kelas	No
1AB	50,90
2AB	10,60
2CV	20,30
1SP	40,70,80

Tabel 2.8 Tabel indeks berdasarkan Jurusan

Jurusan	No
Akuntansi	10,50,90
Manajemen	20,30,60
Teknik Sipil	40,70,80

Indek tidak hanya digunakan untuk pengurutan, tetapi digunakan juga untuk mengakses cepat ke database terutama pencarian data.

Apbila suatu tabel contohnya table Mahasiswa, mengalami perubahan data (penambahan/pengubahan/penghapusan) maka tabel indeks mengalami pengubahan juga.

Application Metadata

Application metadata digunakan untuk menyimpan struktur dan format dari *user forms*, *report*, *queries* dan komponen-komponen aplikasi lainnya.

Konsep Dasar Tabel

Tabel merupakan blok dasar yang paling umum digunakan dalam sistem basis data, atau disebut juga dengan relasi. Komponen tabel terdiri dari beberapa kolom yang ditandai dengan **jenis atribut**. Perpotongan antara baris dan kolom disebut **nilai atribut**. Tujuan penggunaan tabel adalah untuk menyederhanakan logika pandangan terhadap data. Beberapa kententuan-ketentuan dalam penyusunan sebuah tabel adalah sebagai berikut:

a. Urutan baris diabaikan, sehingga pertukaran baris tidak berpengaruh pada isi informasi table.
b. Urutan kolom diabaikan serta identifikasi kolom dibedakan dengan jenis atribut.
c. Tiap perpotongan antara baris dan kolom berisi atribut tunggal.
d. Tiap baris dalam tabel harud dibedakan, sehingga tidak ada dua baris atau lebih dalam table mempunyai nilai atribut yang sama secara keseluruhan.

Tabel yang memenuhi ketentuan ini disebut dengan tabel normal, jika belum maka dilakukan proses normalisasi. Salah satu keuntungan menggunakan basis data adalah konsistensi data selalu terjaga dengan menghindari adanya data rangkap (*redudant data*). Perbedaan antara data rangkap dan data duplikat adalah duplikasi data terjadi bila satu atribut mempunyai dua atau lebih nilai yang sama, sedangkan data rangkap adalah bila satu atribut mempunyai dua atau lebih nilai yang sama, namun bilai salah satu nilai dihapus, maka tidak ada informasi yang hilang, sehingga duplikasi data ini tidak perlu ada. Untuk lebih jelasnya lihat dua tabel berikut:

Tabel 2.9 Contoh duplikasi data pada suatu tabel

NamaPelajar	TeleponPelajar	NamaGuru
Aminudin	7778889	Pardi
Usman	7896532	Pardi
Ari	7474856	Dadang
Rina	7895654	Marni
Tuti	7897744	Dadang
Joni	7845644	Dadang

Tabel 2.10 Contoh adanya kerangkapan data pada suatu table

NamaPelajar	TeleponPelajar	NamaGuru	TeleponGuru
Aminudin	7778889	Pardi	7789665
Usman	7896532	Pardi	7789665
Ari	7474856	Dadang	8965555
Rina	7895654	Marni	4562211
Tuti	7897744	Dadang	8965555
Joni	7845644	Dadang	8965555

Pada tabel 2.9 terjadi duplikasi data pada atribut NamaGuru, andaikan baris pertama pada atribut NamaGuru dihilangkan maka informasi untuk atribut NamaPelajar baris pertama akan hilang, sedangkan pada tabel 2.10 dapat terlihat bahwa kalau atribut TeleponGuru dari baris pertama dihilangkan maka informasi ini masih dapat diketahui melalui atribut NamaGuru pada baris kedua, mengapa? Salah satu syarat tabel normal adalah setiap atribut harus mempunyai nilai tunggal untuk tiap barisnya. Di bawah ini contoh dari suatu tabel yang mempunyai atribut bernilai ganda.

Tabel 2.11 Tabel Dosen dengan nilai ganda

NIP	Nama	Gelar
102	Jackie Ceng	Ir
106	Dakocan	Ir, MSc
503	Ali Oncom	Drs
401	Otoy	Ir, MSc, PhD
203	Gareng	Prof, Drs

Dalam tabel di atas terdapat nilai atribut ganda pada kolom Gelar. Hal ini berakibat pengurutan data hanya dapat dilakukan berdasarkan kolom NIP dan Nama. Untuk menghilangkan nilai ganda tersebut, hal yang paling mudah dilakukan adalah membuat pengisian nilai atribut vertikal namun dapat berakibat kerangkapan data, seperti di bawah ini.

Tabel 2.12 Pengisian atribut secara vertical

NIP	Nama	Gelar
102	Jackie Ceng	Ir
106	Dakocan	Ir
106	Dakocan	MSc
503	Ali Oncom	Drs
401	Otoy	Ir
401	Otoy	MSc
401	Otoy	PhD
203	Gareng	Prof
203	Gareng	Drs

Solusi yang teapat untuk menghilangkan kerangkapan data tersebut adalah dengan membagi table menjadi dua bagian yang saling terhubung dengan elemen penghubung salah satu atributnya. Perhatikan tabel-tabel di bawah ini:

Tabel 2.13 Menghilangkan nilai rangkap dengan pemecahan table

Dosen

NIP	Nama
102	Jackie Ceng
106	Dakocan
503	Ali Oncom
401	Otoy
203	Gareng

Gelar

NIP	Gelar
102	Ir
106	Ir
106	MSc
503	Drs
401	Ir
401	MSc
401	PhD
203	Prof
203	Drs

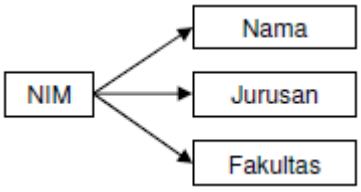
Dengan cara ini dapat mempermudah dalam proses normalisasi berikutnya. Dalam penyusunan aturan data perlu dipahami tentang **determinan** dan **identitas**. Jika sebuah tabel memiliki atribut A, B, dan C, sedangkan A menjadi penentu B atau

sebaliknya B ditentukan oleh A maka A determinan (*functional determines*) B (B *functional dependent* A) . Nilai atribut B dapat saja duplikasi, kosong atau dapat diubah. Jika a1 dan b1 merupakan nilai A maka akan berpasangan dengan nilai B yang sama ataupun berbeda. Jadi A determinan B jika tiap A mempunyai satu pasangan nilai B. Perhatikan contoh tabel di bawah ini:

Tabel 2.14 Tabel Mahasiswa

NIM	Nama	Jurusan	Fakultas
21297956	Asmawi	Teknik Sipil	Teknologi Industri
21297556	Tina	Manajemen	Ekonomi
20399458	Marino	Akuntansi	Ekonomi
21198113	Budi	Teknik Sipil	Teknologi Industri

Apabila setiap nilai atribut NIM menentukan nama mahasiswa maka dikatakan atribut NIM determinan atribut Nama. Begitu juga dengan atribut Jurusan dan Fakultas yang ditentukan oleh NIM. Bentukdiagram determinan adalah sebagai berikut:



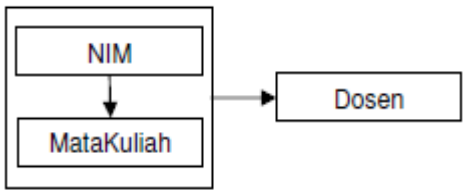
Gambar 2.8 Diagram determinan dari tabel Mahasiswa

Dalam kasus lain, ada kemungkinan dua atribut atau lebih secara bersama menentukan atribut lain atau determinan komposit (*composite determinant/fully functionally dependent*) . Sebagai contoh pada tabel di bawah, atribut NIM dan atribut MataKuliah menentukan atribut Dosen sebagai pengajar.

Tabel 2.15 Tabel pengajaran mata kuliah

NIM	MataKuliah	Dosen
21297956	Matematika Dasar	Pardi
21297956	Fisika Dasar	Munir
20399458	Matematika Dasar	Joko Susilo
21198113	Fisika Dasar	Munir
20399458	Akuntasi Dasar	Marni Siregar
21297556	Matematika Dasar	Joko Susilo

Gambar diagram determinannya adalah sebagai berikut:

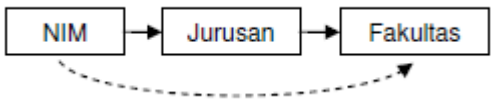


Gambar 2.9 Diagram determinan tabel pengajaran mata kuliah

Sedangkan bila atribut A determinan atribut B dan atribut B merupakan determinan atribut C maka atribut A adalah determinan transitif atribut C (C *transitive dependency* A). perhatikan contoh tabel dan diagram determinan di bawah ini:

Tabel 2.16 Daftar penerimaan mahasiswa baru

NIM	Jurusan	Fakultas
21297956	SI	ILKOM
21297556	SK	ILKOM
20399458	TI	TI
21198113	SK	ILKOM



Gambar 2.10 Diagram A determinan transitif C

Dari pembahasan di atas tiap baris dapat diidentifikasikan dengan semua nilai atribut, tetapi akan sangat menyulitkan. Oleh sebab itu perlu pemilihan salah satu nilai atribut yang digunakan sebagai identitas (*identifier*) atau elemen kunci (*key element*) dari baris. Nilai atribut dapat dijadikan identitas jika dalam tabel tidak terjadi duplikasi data dan data dengan nilai kosong (NULL).

Normalisasi

Proses normalisasi menyediakan cara sistematis untuk meminimalkan terjadinya kerangkapan data diantara relasi dalan perancangan logikal basis data. Format normalisasi terdiri dari lima bentuk, yaitu:

Form Normal Pertama (1NF)

Suatu tabel dikatakan sudah 1NF jika telah memenuhi ketentuan sebagai berikut:

• Tidak ada atribut mempunyai nilai berulang atau nilai array

• Tidak mempunyai baris yang rangkap

Bentuk unnormal mengijinkan nilai-nilai pada suatu atribut dapat berulang. Perhatikan contoh tabeltabel
berikut ini:

Tabel 2.17 Tabel UNIV (University)

DNO	DNAME	DHEAD
EN	English	Lee Kunkel
CS	Computer Science	Albert Roby
MA	Mathematics	Deb Kumar Boy
HS	History	Cathy Doucette
EE	Elecrical Engineering	Raj Chandra Mittra

Tabel 2.18 Tabel INSTR (Instructor)

INAME	IDEG	SPCODE	RANK	SSNO	DNO
Lee Kunkel	BA, MA, PhD	4	Professor	2323121	EN
Albert Roby	BS, MS, PhD	2	Professor	1212154	CS
Deb Kumar Boy	BS, MS, PhD	5	AssocProf	4545454	MA
Cathy Doucette	MA, PhD	6	AssocProf	4545654	HS
Raj Chandra Mittra	BA, MSc, PhD	10	Professor	2231321	EE
Tom Clark	BA, MA	5	AsstProf	1546465	MA
Marcia Brown	BA, BS, MS	2	Instructor	4464654	CS
Susan Woodsmith	MA, MS, PhD	3	AsstProf	2131321	PH
Brady Jackson	MA, DLitt	15	Professor	2456465	RL
Jack Adams	BA, PhD	1	AssocProf	4545462	CS

Tabel 2.19 Tabel STUDNT (Student)

SNAME	SSNO	MAJOR	DEGREE	ADVSR	DNO	COLREG
Roger Brown Smith	121545	Biology	BS	Jack Adams	BI	Arts & Sci.
Cindy Logan	232332	Computer Science	BS	Deb Kumar Boy	MA	Arts & Sci.
Benjamin Johnson	554545	NDEG	NONE		BA	Business
Steve Levin	454545		BA	Lee Kunkel	EN	Arts & Sci.
Tom Jones	899778	Mathematics	MS	Raj Chandra Mittra	EE	Engineering
Berverly Black	365654	English	PhD	Lee Kunkel	EN	Arts & Sci.

Tabel 2.20 Tabel CRSE (Course)

CNO	CNAME	INAME	DNO	SECNO
CS225	Assembler Language	Marcia Brown	CS	02
CS547	Discrete Mathematics	Deb Kumar Roy	CS	01
MA423	Differential Geometry	Tom Clark	MA	04
EN104	English Composition	Staff	EN	04
RL712	Comparative Religion	Brady Jackson	RL	01
CS761	Expert System	Albert Roby	CS	03
EC102	Macroeconomics	Staff	EC	06
EN604	Romanticism	Lee Kunkel	EN	01
HS252	Middle East	Cathy Doucette	HS	02
EE202	Microcomputing	Staff	EE	04
MA611	Algebraic Topology	Tom Clark	MA	01
CS579	Database Systems	Marcia Brown	CS	02
BI104	Biology Concepts	Staff	BI	07

Tabel 2.21 Tabel CRSLST (Course List)

CNO	SECNO	SID	GRADE	OFRNG
CS579	02	121212	A	Spring 87
CS579	02	121231	B-	Spring 87
CS579	02	454549	B+	Spring 87
CS579	02	484545	I	Spring 87
MA611	01	112121	C	Fall 86
MA611	01	212121	A	Fall 86
MA611	01	545454	C+	Fall 86
MA611	01	121215	W	Fall 86

Tabel 2.22 Tabel SPECL (Special)

SPCODE	SPNAME
1	Information Systems
2	Database Systems
3	Kant Doctrine
4	Romantic Literature
5	Differential Geometry
6	Mideast History
7	Topology
8	Automated Reasoning
9	Expert System
10	Microelectronics
11	English Drama
12	Shakespeare
13	Indian History
14	Decision Support Systems
15	Comparatibe Religion

Tabel UNIV, STUDNT, CRSE, CRELST dan SPCODE semuanya berada dalam 1NF, namun untuk tabel INSTR masih dalam keadaan unnormalisasi, karena atribut IDEG menerima nilai array seperti “BA, MA, PhD” atau “MA, MS, PhD”.

Form Normal Kedua (2NF)

Relasi dapat dikatakan format normal kedua jika sudah dalam format normal pertama dan diikuti kondisi sebagai berikut:

• Key terdiri dari atribut tunggal

• Setiap atribut nonkey ketergantungan fungsional pada semua key atau tidak terjadinya ketergantungan pada key *composite*.

Misalnya tabel UNIV berada dalam normal kedua dengan mengasumsikan DNO sebagai key, kecuali CRSE. Jika ditentukan CNO dan SECNO sebagai key composite, atribut nonkey CNAME tergantung hanya pada CNO, bukan pada SECNO, sehingga CNAME tidak secara ketergantungan fungsional penuh terhadap key (CNO, SECNO).

Form Normal Ketiga (3NF)

Relasi dikatakan format normal ketiga jika sudah dalam format normal kedua dan tidak ada *ketergantungan transitif* diantara atribut. Misalnya tabel STUDNT mempunyai atribut SSNO sebagai key (2NF). *Ketergantungan transitif* terjadi diantara DNO dan COLREG. Saat DNO determinan COLREG tanpa melibatkan key SSNO. Contohnya, DNO=’CS’ termasuk COLREG=’Arts/Sc.’ tidak tergantung oleh atribut SSNO, sehingga STUDNT belum termasuk 3NF. Yang menjadi catatan, *ketergantungan transitif* tidak akan terjadi jika ada *ketergantungan fungsional* diantara atribut-atribut nonkey yang melibatkan key. Misalnya atribut nonkey SNAME determinan atribut nonkey lainnya yaitu MAJOR, DEGREE, ADVSR dan DNO. Tetapi hal ini merupakan *ketergantungan fungsional* bukan *ketergantungan transitif* selagi semua melibatkan key SSNO.

Form Normal Boyce-Codd (BCNF)

BCNF menentukan setiap determinan adalah kunci kandidat (*candidate key*). Misalnya UNIV mempunyai dua determinan yaitu DNO dan DNAME yang merupakan *kunci kandidat* sehingga termasuk ke dalam BCNF. Di lain pihak CRSLST dalam 3NF tetapi tidak dalam BCNF. Atribut komposisinya (CNO, SECNO, SID, OFRNG) sebagai kunci-kunci kandidat dan tidak ada ketergantungan transitif, sehingga CRSLST termasuk ke dalam 3NF. Namun atribut CNO adalah determinan saat SECNO tergantung penuh secara fungsional terhadap CNO, walaupun CNO bukan kunci kandidat, sehingga CRSLST belum termasuk BCNF.

Form Normal Keempat (4NF)

Bentuk ini adalah bentuk normal ketiga atau BCNF dengan nilai atribut tidak tergantung pada nilai banyak (*multivalue dependency*).

Form Normal Kelima (5NF)

Konsep pada bentuk ini adalah ketergantungan pada gabungan beberapa atribut (*join dependency*). Bentuk lain proses normalisasi dapat anda lihat dalam tabel-tabel di bawah ini:

Tabel 2.23 Tabel Mahasiswa dalam 1NF

NIM	NamaAwal	NamaAkhir	Fakultas
122233	Asep	Darma	Ilmu Komputer
233323	Angling	Darma	Ilmu Komputer
244455	Bergola	Ijo	Hukum
334343	Jaka	Sembung	Kebidanan
322323	Jaka	Tarub	Hukum

Pada tabel di atas berada pada 1NF kerana tidak ada baris yang duplikat, setiap kolom hanya mempunyai nilai tunggal (tidak ada group berulang atau array) dan semua masukan dalam kolom mempunyai jenis yang sama. Key NIM secara fungsional menentukan atribut lain seperti NamaAwal, NamaAkhir, dan Fakultas (dengan asumsi, setiap mahasiswa hanya boleh menempatkan satu fakultas). NIM®NamaAwal, NIM®NamaAkhir, NIM®Fakultas. Atribut key menentukan secara unik nilai dari atribut lain dalam tabel, semua atribut non-key dalam tabel secara fungsional tergantung terhadap key. Tetapi ada kemungkinan atribut non-key dalam table dapat menentukan atribut lain pada tabel tersebut. Perhatikan tabel berikut:

Tabel 2.24 Atribut Non-Key Determinan

NamaAwal	NamaAkhir	Fakultas	Jenjang
Asep	Darma	Ilmu Komputer	S1
Angling	Darma	Ilmu Komputer	S1
Bergola	Ijo	Hukum	S1
Jaka	Sembung	Kebidanan	D3
Jaka	Tarub	Hukum	S1

Atribut jenjang dapat dikatakan tergantung secara fungsional pada atribut Fakultas dengan konsekuensi bahwa Fakultas Ilmu Komputer dan Hukum hanya untuk mahasiswa S1 dan Fakultas Kebidanan hanya untuk mahasiswa D3.

Pada tabel 2.23 juga memiliki *composite key* (kunci komposisi) yang terdiri dari atribut NamaAwal dan NamaAkhir dan atribut Jenjang tergantung secara fungsional pada *composite key* tersebut. Sebenarnya bisa saja atribut Jenjang tergantung pada atribut NamaAkhir (jika nilai atribut NamaAkhir tidak ada yang sama, namun NamaAkhir terdapat nilai yang sama yaitu "Darma"), atau tergantung dengan atribut NamaAwal, tetapi atribut NamaAwal mempunyai nilai duplikat yaitu "Jaka". Maka atribut Jenjang tidak tergantung fungsional terhadap kedua atribut key tersebut. Sehingga table tersebut belum termasuk ke dalam 2NF (semua atribut non-key tergantung pada semua key). Tabel 2.23 dapat dinormalisasikan ke dalam bentuk 2NF jika di tambahkan atribut NIM, sehingga table hanya mempunyai atribut tunggal (non-composite key) seperti di bawah ini.

Tabel 2.25 Tabel Normalisasi 2NF

NIM	NamaAwal	NamaAkhir	Fakultas	Jenjang
122233	Asep	Darma	Ilmu Komputer	S1
233323	Angling	Darma	Ilmu Komputer	S1
244455	Bergola	Ijo	Hukum	S1
334343	Jaka	Sembung	Kebidanan	D3
322323	Jaka	Tarub	Hukum	S1

Anomali (Pelanggaran) dan Normalisasi

Untuk mengilustrasikan adanya anomali, anggap terjadi pada atribut Jenjang dari Fakultas, misalnya Kebidanan. Jika Jaka Sembung kembali berkelana (tidak jadi kuliah) maka baris yang berhubungan dengan Jaka Sembung akan dihapus dan terjadi kehilangan informasi bahwa Kebidanan mempunyai jenjang D3. Kita dapat saja mengetahui bahwa Jenjang Kebidanan adalah D3, tetapi dalam database tersebut tidak ada informasi yang menyatakan Jenjang Kebidanan.

Contoh lain misalnya penambahan mahasiswa baru yang bernama Suparman yang ingin kuliah di Fakultas Teknik. Fakta dari informasi tabel kita tidak dapat mengetahui apa jenjang dari Fakultas Teknik dan juga apakah Fakultas Teknik terdapat pada Universitas yang bersangkutan. Dari hal ini kita tidak akan mengetahui apakah dapat nilai Teknik tersebut dapat dimasukkan ke dalam atribut Fakultas atau tidak. Apabila dapat dimasukkan ke atribut Fakultas, bagaimana dengan atribut Jenjang-nya?

DBMS hanya dapat bekerja dengan informasi yang terdapat pada tabel-tabel dan aturan-aturan yang bekerja pada tabel-tabel tersebut dengan tepat dan mungkin. Bagaimana hubungan antara anomali dan normalisasi? Jawaban yang singkat adalah dengan menyusun tabel-tabel dalam database cukup dinormalkan (dalam praktek umumnya sampai normal keempat), dan menjamin bahwa anomali tidak terjadi pada database.

Porses normalisasi kelihatan sangat menyulitkan, ketika melihat dari definisi tiap-tiap tingkatan normalisasi. Namun dalam prakteknya kita dapat mencapainya dengan menjamin bahwa tabel-tabel terdiri dari tabel “single-theme”.

Walaupun dalam 2NF dapat terjadi penambahan maupun penghapusan data yang mengakibatkan anomali, kita dapat membentuk tabel tersebut menjadi beberapa tabel “**single-theme**”. Gagasan ini dapat diilustrasikan pada tabel di bawah ini.

Tabel 2.26 Tabel-tabel "Single-Theme"

NIM	NamaAwal	NamaAkhir
122233	Asep	Darma
233323	Angling	Darma
244455	Bergola	Ijo
334343	Jaka	Sembung
322323	Jaka	Tarub

Fakultas	Jenjang
Ilmu Komputer	S1
Hukum	S1
Kebidanan	D3

NIM	Fakultas
122233	Ilmu Komputer
233323	Ilmu Komputer
244455	Hukum
334343	Kebidanan
322323	Hukum

Transitive Dependencies (Ketergantungan transitif) (3NF)

Seperti yang telah kita ketahui, ketergantungan transitif terjadi bilamana suatu A®B dan B®C, maka A®C. Ilustrasi untuk kejadian ini dapat dilihat pada tabel berikut.

Tabel 2.27 Tabel Daftar Buku

Author Last Name	Author First Name	Book Title	Subject	Collection or Library	Building
Berdahl	Robert	The Politics of the Prussian Nobility	History	PCL General Stacks	Perry-Casta Library
Yudof	Mark	Child Abuse and Neglect	Legal Procedures	Law Library	Townes Hall
Harmon	Glynn	Human Memory and Knowledge	Cognitive Psychology	PCL General Stacks	Perry-Casta Library
Graves	Robert	The Golden Fleece	Greek Literature	Classics Library	Waggener Hall
Miksa	Francis	Charles Ammi Cutter	Library Biography	Library and Information Science Collection	Perry-Casta Library
Hunter	David	Music Publishing and Collecting	Music Literature	Fine Arts Library	Fine Arts Building
Graves	Robert	English and Scottish Ballads	Folksong	PCL General Stacks	Perry-Casta Library

Dari tabel di atas kita dapat berpendapat bahwa buku yang ber-subyek History, cognitive psychology, dan folksong diberikan ke PCL General Stacks collection; sedangkan Legal procedure diberikan ke Law Library; Greek Literature diberikan ke Classic Library; sedangkan Library Biography diberikan ke Library and Information Science Collection; dan Music Literature diberikan ke Fine Arts Library.

Kemudian kita dapat menduga bahwa PCL General Stacks Collection dan LISC keduanya ditempatkan di gedung Perry-Casta Library (PCL); Classic Library ditempatkan di gedung Waggener Hall; Law Library and Fine Arts Library di tempatkan di gedung Townes Hall dan Fine Arts Building. Sehingga kita dapat melihat ketergantungan transitif pada tabel di atas, dimana buku-buku History, Cognitive Psychology atau Library Biography secara fisik ditempatkan di gedung PCL; buku Lega Procedures di tempatkan di Townes Hall dan begitu seterusnya.

Apa yang salah ketika terjadinya ketergantungan transitif pada tabel tersebut? Hal pertama terjadinya duplikasi informasi, dimana tiga baris berbeda mengacu pada PCL General Stacks yang berada di gedung PCL. Kesalahan yang kedua adalah kemungkinan adanya penghapusan anomali, yaitu jika kita hapus baris dengan nama pengarangYudof maka kita akan kehilangan informasi Law Library di dalam Townes Hall. Ketiga dapat pula terjadi penambahan anomali jika kita akan menambah buku Chemistry, tetapi kenyataannya tidak ada data yang menyatakan Chemistry Library berada di gedung Robert A. Welch Hall. Dan masalah keempat terjadinya kesalahan meng-update jika pegawai memasukan buku ke LISC tetapi memasukkan Townes Hall ke atribut Building. Solusi untuk masalah di atas adalah membentuk tabel tersebut menjadi tabel-tabel “single-theme”.

Tabel 2.28 Tabel-tabel "single-theme " untuk tabel transitive dependencies

Author Last Name	Author First Name	Book Title
Berdahl	Robert	The Politics of the Prussian Nobility
Yudof	Mark	Child Abuse and Neglect
Harmon	Glynn	Human Memory and Knowledge
Graves	Robert	The Golden Fleece
Miksa	Francis	Charles Ammi Cutter
Hunter	David	Music Publishing and Collecting
Graves	Robert	English and Scottish Ballads

Book Title	Subject
The Politics of the Prussian Nobility	History
Child Abuse and Neglect	Legal Procedures
Human Memory and Knowledge	Cognitive Psychology
The Golden Fleece	Greek Literature
Charles Ammi Cutter	Library Biography
Music Publishing and Collecting	Music Literature
English and Scottish Ballads	Folksong

Subject	Collection or Library
History	PCL General Stacks
Legal Procedures	Law Library
Cognitive Psychology	PCL General Stacks
Greek Literature	Classics Library
Library Biography	Library and Information Science Collection
Music Literature	Fine Arts Library
Folksong	PCL General Stacks

Collection or Library	Building
PCL General Stacks	Perry-Casta Library
Law Library	Townes Hall
PCL General Stacks	Perry-Casta Library
Classics Library	Waggener Hall
Library and Information Science Collection	Perry-Casta Library
Fine Arts Library	Fine Arts Building
PCL General Stacks	Perry-Casta Library

Kita dapat lihat bahwa semua tabel tidak mempunyai ketergantungan transitif (3NF, Domain Key Normal Form (DKNF)).

Untuk tabel pengarang terdapat nama depan yang sama yaitu Robert, dalam hal ini kita menyarankan untuk membuat atribut identifikasi dalam menyatakan nama pengarang, seperti di bawah ini.

Tabel 2.29 Tabel pengarang setelah penambahan atribut ID

Author Last Name	Author First Name	ID Author
Berdahl	Robert	001
Yudof	Mark	002
Harmon	Glynn	003
Graves	Robert	004
Miksa	Francis	005
Hunter	David	006
Graves	Robert	007

ID Author	Book Title
001	The Politics of the Prussian Nobility
002	Child Abuse and Neglect
003	Human Memory and Knowledge
004	The Golden Fleece
005	Charles Ammi Cutter
006	Music Publishing and Collecting
007	English and Scottish Ballads

Pembentukan tabel-tabel di atas akan lebih menghemat kapasitas media penyimpan danbmeminimalkan kesalahan dalam pemasukkan data yang berupa key. BCNF mengijinkan terjadinya anomali ketika tabel gagal memiliki properti yaitu setiap determinan adalah kunci kandidat (*candidate key*). Contoh pada tabel di bawah ini gagal memiliki properti ini. Dalam tabel ini SSN ditafsirkan sebagai pelajar dengan Major serta Adviser (pembimbing). Dengan catatan tiap pelajar pelajar 123-45-6789 dan 987-65-4321 mempunyai dua jurusan dengan pembimbing yang berbeda tiap jurusannya.

Tabel 2.30 Tabel Mahasiswa dengan BCNF

SSN	Major	Adviser
123-45-6789	Library and Information Science	Dewey
123-45-6789	Public Affairs	Roosevelt
222-33-4444	Library and Information Science	Putnam
555-12-1212	Library and Information Science	Dewey
987-65-4321	Pre-Medicine	Semmelweis
987-65-4321	Biochemistry	Pasteur
123-54-3210	Pre-Law	Hammurabi

Dalam tabel di atas salah satu determinan adalah atribut berpasangan yaitu SSN dan Major. Tiap pasangan nilai atribut SSN dan Major menentukan nilai unik untuk atribut Adviser. Determinan lain adalah pasangan SSN dan Adviser yang menentukan nilai unik atribut Major. Determinan lain lagi adalah Adviser dimana setiap nilai adviser yang berbeda menentukan secara unik nilai atribut Major. (catatan: setiap pelajar mempunyai pembimbing (adviser) tunggal untuk setiap jurusannya (major) dan tiap pembimbing (adviser) membimbing hanya satu jurusan)

Sekarang kita akan menguji ketiga determinan tersebut apakah sebagai kunci kandidat. Pasangan SSN dan Major adalah kunci kandidat untuk setiap pasangan unik mengidentifikasikan sebuah baris dalam tabel. Pasangan SSN dan Adviser juga kunci kandidat. Tetapi determinan Adviser bukan sebagai kunci kandidat karena nilai Dewey terdapat dalam dua baris pada kolom Adviser. Jadi tabel diatas tidak dapat dikatakan dalam kondisi setiap determinan adalah kunci kandidat.

Sangatlah mudah untuk memeriksa anomali dalam tabel ini. Misalnya jika pelajar 987-65-4321 meninggalkan universitas tersebut, tabel akan kehilangan informasi bahwa Semmelweis adalah seorang pembimbing untuk jurusan Pre-Medicine. Contoh lain adalah tidak adanya informasi tentang pembimbing-pembimbing untuk pelajar dengan jurusan History (Sejarah).

Solusi untuk masalah di atas adalah membuat tabel tersebut dipecah menjadi beberapa tabel-tabel single-theme seperti berikut:

Tabel 2.31 Tabel Single theme

SSN	Adviser
123-45-6789	Dewey
123-45-6789	Roosevelt
222-33-4444	Putnam
555-12-1212	Dewey
987-65-4321	Semmelweis
987-65-4321	Pasteur
123-54-3210	Hammurabi

Major	Adviser
Library and Information Science	Dewey
Public Affairs	Roosevelt
Library and Information Science	Putnam
Pre-Medicine	Semmelweis
Biochemistry	Pasteur
Pre-Law	Hammurabi
History	Herodotus

Maka kedua tabel di atas dalam kondisi BCNF yaitu setiap determinan merupakan kunci kandidat.

Bentuk formal keempat memperhatikan terjadinya anomali ketika tabel gagal memiliki properti bahwa tidak ada ketergantungan nilai banyak (multivalued). Contoh tabel di bawah merupakan bentuk anomali untuk ketergantungan dengan nilai banyak.

Tabel 2.32 Mahasiswa dengan daftar hobi

LastName	Major	Hobby
Jones	Library and Information Science	Surfing the Internet
Jones	Library and Information Science	Chess
Jones	Public Affairs	Surfing the Internet
Jones	Public Affairs	Chess
Lee	Library and Information Science	Photography
Lee	Library and Information Science	Stamp collecting
Ruiz	Pre-Medicine	Surfing the Internet
Ruiz	Pre-Medicine	Photography
Ruiz	Biochemistry	Surfing the Internet
Ruiz	Biochemistry	Photography
Smith	Pre-Law	Playing poker

Anggap pelajar Jack Jones hobinya menjelajah internet dan bermain catur; Lynn Lee hobinya photographer dan koleksi perangko; Mary Ruiz, hobinya menjelajah internet dan photographer; dan Lynn Smith, hobinya bermain poker.

Situasi ini dapat mengakibatkan redudansi data dan memberikan efek ketergantungan nilai banyak terjadi ketika (a) tabel sekurangnya mempunyai tiga atribut (b) dua atribut mempunyai nilai banyak dan (c) nilai dari atribut multivalued tergantung hanya satu dari sisa atribut. Atribut LastName menentukan nilai banyak dari atribut Major dan Hobby, tetapi keduanya tidak saling tergantung terhadap yang lainnya (independent).

Notasi untuk ketergantungan multivalue adalah panah ganda. Untuk contoh ini kita dapat menulisnya dengan bentuk LastName - Major dan LastName - Hobby (LastName multideterminan Major dan LastName Multideterminan Hobby).

Tabel 2.33 Bentuk Tabel-Tabel yang Dijadikan Single-Theme

LastName	Major
Jones	Library and Information Science
Jones	Public Affairs
Lee	Library and Information Science
Ruiz	Pre-Medicine
Ruiz	Biochemistry
Smith	Pre-Law

LastName	Hobby
Jones	Surfing the Internet
Jones	Chess
Lee	Photography
Lee	Stamp collecting
Ruiz	Surfing the Internet
Ruiz	Photography
Smith	Playing poker

Normal Form Kelima sulit untuk diilustrasikan dengan contoh sederhana. Karenanya di sini tidak meng-ilustrasikan properti dari 5NF mempunyai tiap *join dependency* dalam tabel menjadi penyebab membentuk kunci kandidat tabel. Alasan lain yang pertama, dalam prakteknya 4NF sudah mencukupi dan yang kedua adalah Domain-Key Normal Form (DKNF) sudah termasuk 5NF.

DKNF digunakan sebagai solusi untuk menghindari anomali: kumpulan tabel-tabel (relasi) dalam DKNF sebagai konsekuensi dari teori Ronald Fagin (1981) untuk bebas dari anomali. Definsi dari DKNF adalah relasi dalam DKNF jika setiap *constrain* relasi logikal mendefinisikan kunci dan domain. Menurut Fagin istilah kunci adalah primary key dan candidate key, domain merupakan kumpulan definisi dari isi atribut (kolom) dan beberapa batasan jenis data yang akan disiman dalam kolom, sebagai contoh batasan hanya data numerik atau hanya data logikal. Constraint berarti aturan-aturan terhadap atribut dengan jelas sehingga dapat

diputuskan aturan yang mana dijalankan atau dilanggar untuk kumpulan data yang mana diterima. Sebagai ilustrasi ini simak tabel di bawah ini:

Konversi tabel dengan partial dependency ke tabel DKNF

Tabel 2.34 Tabel dengan partial dependency

FirstName	LastName	Major	Level
Jack	Jones	LIS	Graduate
Lynn	Lee	LIS	Graduate
Mary	Ruiz	Pre-Medicine	Undergraduate
Lynn	Smith	Pre-Law	Undergraduate
Jane	Jones	LIS	Graduate

Dalam tabel di atas terdapat komposisi kunci yang terdiri dari pasangan atribut LastName-FirstName, dan semua atribut tergantung dengan kunci ini. Tetapi ada hal lain yang penting: atribut Level tergantung pada atribut LastName, Level hanya tergantung pada sebagian kunci. Sehingga tabel ini belum dikatakan DKNF.

Kita akan membutuhkan tabel yang menyediakan link antara pasangan atribut FirstName dan LastName dan atribut Major. Dalam tabel tersbeut Major menjadi konsekuensi logikal dari key dan domain. Sehingga dibutuhkan dua tabel, satu berisi Major dan Level dan yang lain FirstName, LastName dan Major.

Contoh tabel-tabel di bagian sebelumnya sudah termasuk DKNF. Jadi dengan membentuk table menjadi single-theme hampir sama halnya dengan membentuk tabel DKNF.

PEMODELAN BASIS DATA

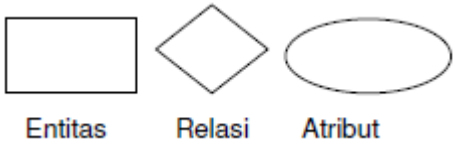
Model Hubungan Antar Entitas (Entity Relationship-Model)

Model entity-relationship pertama kali diperkenalkan oleh Peter Chen pada tahun 1976. Dalam pemodelan ini dilakukan dengan tahapan sebagai berikut:

a. Memilih entitas-entitas yang akan disusun dalam basis data dan menentukan hubungan antar entitas yang telah dipilih.

b. Melengkapi atribut-atribut yang sesuai pada entitas dan hubungan sehingga diperoleh bentuk tabel normal penuh (ternormalisasi).

Elemen-elemen dalam model ER dapat digambarkan pada gambar diagram di bawah ini:



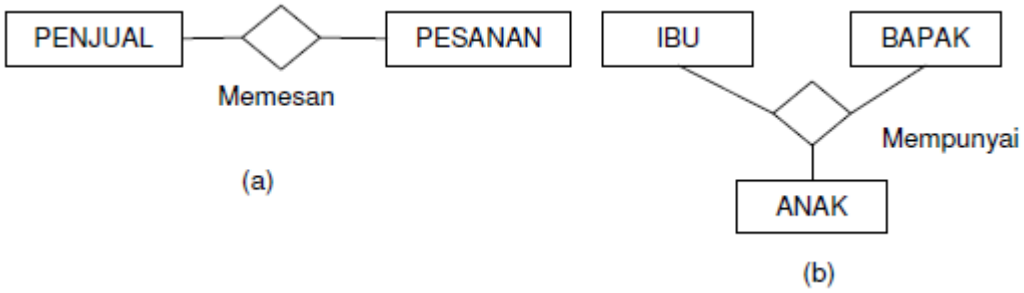
Gambar 2.11 Elemen-elemen ER-Model

Entitas merupakan sesuatu yang dapat diidentifikasikan dalam lingkungan kerja pengguna. Entitas yang diberikan tipe dikelompokkan ke kelas entitas. Perbedaan antara kelas entitas dan instansi entitas adalah sebagai berikut:

- Kelas entitas adalah kumpulan entitas dan dijelaskan oleh struktur atau format entitas di dalam kelas.
- Instansi kelas merupakan bentuk penyajian dari fakta entitas.

Umumnya terdapat banyak instansi entitas di dalam setiap entitas kelas. Setiap entitas kelas memiliki atribut yang menjelaskan karakteristik dari entitas tersebut, sedangkan setiap instansi entitas mempunyai identifikasi yang dapat bernilai unik (mempunyai nilai yang berbeda untuk setiap identifikasinya) atau non-unik (dapat bernilai sama untuk setiap identifikasinya).

Antara entitas diasosiakan dalam suatu hubungan (*relationship*). Suatu relasi dapat memiliki beberapa atribut. Jumlah kelas entitas dalam suatu relasi disebut **derajat relasi**. Gambar di bawah ini merupakan contoh dari relasi berderajat dua dan relasi berderajat tiga.

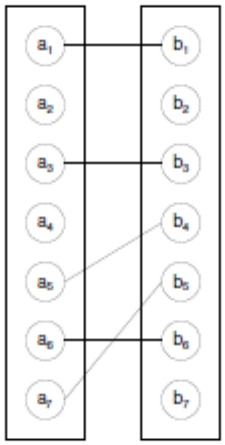


Gambar 2.12 (a) Relasi berderajat dua (b) Relasi berderajat tiga

Tipe Binary Relationship

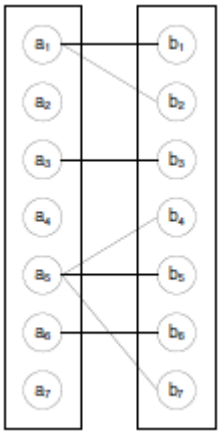
Relasi memiliki tiga tipe biner yaitu:

a. One-to-one (1:1). Hubungan terjadi bila setiap instansi entitas hanya memiliki satu hubungan dengan instansi entitas lain.



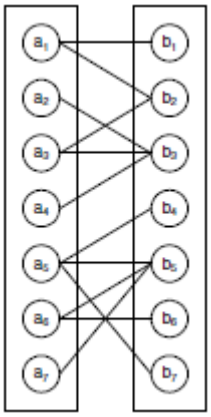
Gambar 2.13 Hubungan 1:1 (one-to-one)

b. One-to-many (1:M). Relasi ini terjadi bila setiap instansi entitas dapat memiliki lebih dari satu hubungan terhadap instansi entitas lain tetapi tidak kebalikannya.



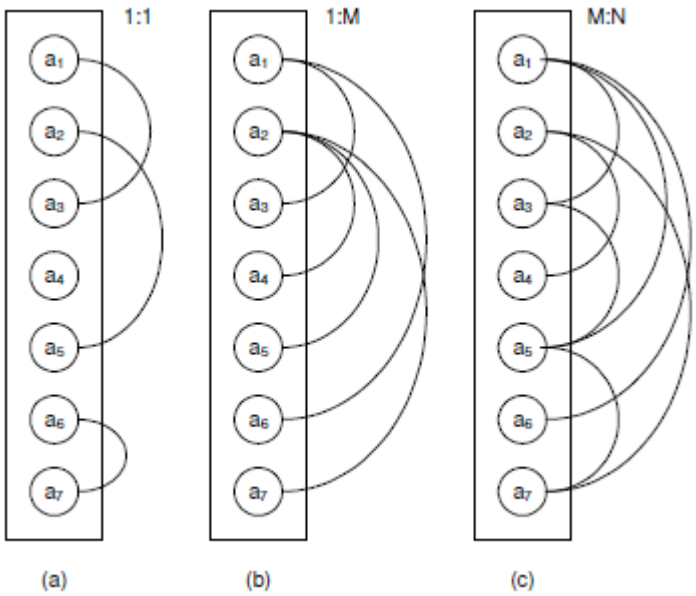
Gambar 2.14 Hubungan 1:M (one-to-many)

d. Many-to-many (M:N). Hubungan saling memiliki lebih dari satu dari setiap instansi entitas terhadap instansi entitas lainnya.



Gambar 2.15 Hubungan M:N (many-to-many)

Selain relasi antara dua entitas, terdapat juga hubungan terhadap entitasnya sendiri yang disebut dengan *recursive relationship* (*self relation*). Hubungan ini dapat mempunyai tipe biner, seperti yang sudah dibahas sebelumnya.

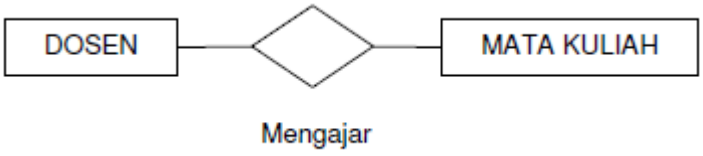


Gambar 2.16 (a) Relasi rekursif 1:1 (b) Relasi rekursif 1:M (c) Relasi rekursif M:N

Partisipasi Hubungan

Partisipasi atau keterlibatan setiap instansi entitas dalam membentuk hubungan dapat bersifat wajib (obligatory/digambarkan dengan garis penuh) atau tidak wajib (non-obligatory/digambarkan dengan garis putus-putus) dalam aturan data. Misalnya hubungan antara DOSEN dan Mata Kuliah dengan aturan data sebagai berikut:

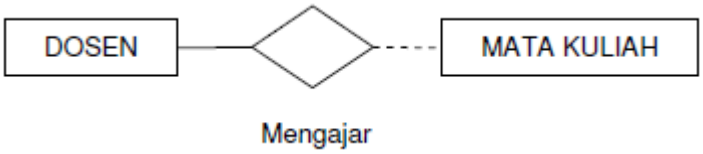
Setiap dosen harus mengajar satu mata kuliah dan setiap mata kuliah harus diajarkan oleh seorang dosen. Setiap anggota entitas dalam hubungan adalah wajib untuk instansi entitas DOSEN maupun Mata Kuliah.



Gambar 2.17 Diagram E-R dengan partisipasi wajib

Dalam bentuk aturan lain yaitu:

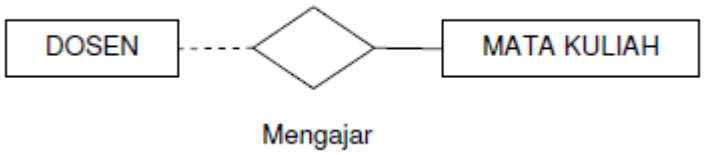
Setiap dosen harus mengajar satu mata kuliah dan setiap mata kuliah mungkin diajarkan oleh seorang dosen atau mungkin tidak sama sekali.



Gambar 2.18 Diagram E-R dengan partisipasi wajib pada sisi Dosen dan tidak wajib pada sisi Mata Kuliah

Atau aturan lain:

Setiap dosen hanya boleh mengajar satu mata kuliah atau boleh saja dosen tidak mengajar dan setiap mata kuliah harus diajarkan oleh seorang dosen.



Gambar 2.19 Diagram E-R dengan partisipasi tidak wajib pada sisi Dosen dan wajib pada sisi Mata Kuliah

Aturan lain yang tidak mewajibkan kedua belah pihak:

Setiap dosen hanya boleh mengajar satu mata kuliah atau boleh saja dosen tidak mengajar dan setiap mata kuliah hanya boleh diajarkan oleh seorang dosen atau tidak sama sekali.



Gambar 2.20 Diagram E-R dengan partisipasi tidak wajib pada kedua sisi

PERANCANGAN BASIS DATA

Permasalahan dalam perancangan basis data adalah *bagaimana merancang struktur logikal dan fisikal dari satu atau lebih basis data untuk memenuhi kebutuhan informasi yang diperlukan oleh pengguna sesuai dengan aplikasi-aplikasi yang ditentukan.*

Dengan permasalahan tersebut dapat ditentukan beberapa tujuan utama perancangan basis data,

yaitu:

a. Memenuhi kebutuhan informasi sesuai dengan yang diperlukan oleh pengguna untuk aplikasi tertentu.
b. Mempermudah pemahaman terhadap struktur informasi yang tersedia dalam basis data,
c. Memberikan keterangan tentang persyaratan pemrosesan dan kemampuan sistem, seperti lama tidaknya mengakses data, kapasitas memori yang tersedia dan sebagainya.

Tahapan-tahapan proses perancangan untuk memenuhi tujuan tersebut adalah:

1. Mengumpulkan dan menganalisis persyaratan
2. Merancang konsepsual basis data
3. Memilih Sistem Manajemen Basis Data
4. Merancang logikal basis data
5. Merancang fisikal basis data (pemetaan model data)
6. Implementasi sistem basis data

Dalam pelaksanaan perancangan tersebut terdapat dua kegiatan yang dapat dilakukan secara paralel, yaitu perancangan struktur dan isi data (analisis data) dan perancangan pemrosesan data serta program aplikasi (analisis fungsional). Tahapan rancangan basis data seperti pada bagan di bawah ini tidak secara ketat harus diikuti secara berurutan. Karena antara tahap yang satu dengan yang lainnya dapat saling mempengaruhi dan memberi umpan balik.

Rancangan konsepsual basis data (tahap 2) menghasilkan skema konsepsual dari basis data yang bebas dari DBMS tertentu. Dalam hal ini juga digunakan

pemodelan bahasa tingkat tinggi seperti model E-R (*Entity Relationship*) atau EER (*Enhanced Entity Relationship*). Tahap ini juga menentukan transaksi data yang dapat dilakukan terhadap sistem basis data. Rancangan logikal (tahap 4) disebut juga pemetaan model data, yaitu mentransformasikan model data yang telah dibuat pada tahap dua ke dalam model data yang sesuai dengan DBMS terpilih. Tahap ini juga melakukan perancangan skema eksternal untuk aplikasi yang ditentukan.

Rancangan fisikal basis data (tahap 5) melakukan pendefinisian basis data yang akan disimpan sesuai dengan SMBD yang digunakan, meliputi struktur penyimpanan data, format data dan jalurakses. Tahap ini disebut skema internal.

Koleksi dan Analisis Persyaratan

Koleksi dan analisis persyaratan merupakan proses pengumpulan dan analisis tujuan dan harapan pengguna untuk memperoleh informasi dari sistem basis data. Kegiatan-kegiatan yang dilakukan dalam tahapan ini adalah sebagai berikut:

a. Melakukan identifikasi bidang aplikasi dan kelompok pemakai.

b. Mempelajari dan menganalisis dokumen yang ada pada aplikasi tertentu.

c. Mempelajari sistem yang sedang berjalan.

d. Membuat semacam pertanyaan/angket pada calon pengguna yang dipandang potensial untuk memperoleh spesifikasi informasi dan proses yang diperlukan.

Perancangan Konsepsual Basis Data

Tahapan ini meliputi dua kegiatan yaitu rancangan skema konsepsual tentang organisasi data yang harus disimpan dalam basis data, dan rancangan transaksi yang dilakukan untuk memperoleh informasi dari sistem basis data hasil analisis persyaratan pada tahap 1.

Rancangan skema konsepsual

Hasil rancangan konsepsual merupakan pemodelan data dari pemahaman dunia nyata yang dituliskan dalam bahasa tingkat tinggi dan tidak terikat dengan DBMS yang akan digunakan. Umumnya pembuatan skema konsepsual ini menggunakan diagram

E-R. Untuk menyusun rancanngan konsepsual dimulai dengan identifikasi komponen utama dari skema (entitas, hubungan, atribut) dengan mengacu pada karakateristik sebagai berikut:

a. Model data harus cukup memberikan tampilan yang menggambarkan perbedaan jenis data, hubungan dan *constraint* (ekspresif).

b. Model harus dibuat sederhana dan mudah dipahami serta digunakan oleh pengguna.Penyajian model data dibuat dalam diagram yang mudah diinterprestasi.

c. Penyajian model data dalam skema harus teliti dan tidak menimbulkan interprestasi (akurat).

Rancangan transaksi

Teknik pembuatan spesifikasi transaski dilakukan dengan melakukan identifikasi data masukan dan data keluaran serta sifat fungsional transaksi, sehingga perancang dapat membuat model konsepsual transaksi yang tidak terikat dengan sistem.

Fungsi-sungsi model transaksi adalah sebagai berikut:

a. Transaksi pemanggilan (*retrieval transaction*), yaitu pemanggilan data untuk ditampilkan di layar monitor atau dicetak sebagai laporan.

b. Transaksi pembaharuan (*update transaction*), digunakan untuk pemasukan data baru atau perubahan data lama.

c. Transaksi campuran (*mixed transaction*), digunakan untuk kombinasi pemanggilan data dan pembaharuan data.

Pemilihan DBMS

Faktor-faktor yang menentukan pemilihan DBMS antara lain adalah faktor teknik, faktor ekonomi dan politik dalam organisasi. Faktor teknik meliputi kelangsungan dari DBMS untuk diterapkan dalam pengelolaan data seprti jenis model DBMS, struktur penyimpanan data dan alur akses data, interface pengguna dan pemrogram, jenis bahasa tingkat tinggi dan sebagainya. Faktor ekonomi

diantaranya pembelian software DBMS, pembelian hardware, biaya pemeliharaan sistem, biaya penyusunan basis data dan lain sebagainya.

Pemodelan Logikal Basis Data

Tujuan dari tahap ini adalah menyusun rancangan konsepsual dan skema eksternal yang sesuai
dengan DBMS yang dipilih. Langkah-langkah yang dilakukannya adalah:
a. Pemetaan (transformasi data) yang tidak terikat sistem.
b. Penyusunan skema sesuai dengan DBMS.

Perancangan Fisikal Basis Data

Tujuan dari perancangan ini adalah untuk membuat spesifikasi struktur penyimpanan dan jalur akses
data sehingga diperoleh kemampuan sistem yang baik untuk berbagi aplikasi.
Beberapa hal yang menjadi pertimbangan dalam perancangan fisikal adalah:

1. Waktu tanggap. Yaitu waktu yang digunakan oleh sistem sejak transaski basis data dimasukkan untuk dieksekusi sampai mendapat tanggapan dari sistem. Faktor yang mempengaruhinya dalah waktu akses basis data yang dikendalikan oleh DBMS serta dipengaruhi oleh sistem pemuatan data (*loading*) pada komputer, sistem operasi yang digunakan, atau penundaan system komunikasi.

2. Penggunaan memori komputer. Merupakan kapasitas memori komputer yang digunakan untuk menyimpan berkas-berkas basis data dan struktur jalur akses.

3. Transaksi data. Kemampuan melakukan transaksi data tiap satuan waktu merupakan hal yangkritis.

Implementasi Sistem Basis Data

Tahap ini merupakan implementasi dari hasil pemodelan logikal dan fisikal. Bahasa yang digunakan untuk definisi data atau penyimpanan data yang sesuai dengan DBMS terpilih. Implementasi penyusunan basis data dimulai dari pembuatan

berkas-berkas data kosong yang akan digunakan untuk menyimpan data dalam basis data. Kemudian dilanjutkan dengan pemasukan data untuk tiap instansi tabel.

Dalam impelementasi rancangan transaksi, program aplikasi ditulis dengan bahasa manipulasi data yang sesuai. Program-program aplikasi yang dibuat harus dilakukan uji coba dulu untuk menguji kebenaran program. Setelah diuji kemudian diimplementasikan dalam operasional sistem basis data.

2.2.3 Jaringan komputer

Pengertian

Jaringan komputer adalah sebuah kumpulan komputer, printer dan peralatan lainnya yang terhubung. Informasi dan data bergerak melalui kabel-kabel sehingga memungkinkan pengguna jaringan komputer dapat saling bertukar dokumen dan data, mencetak pada printer yang sama dan bersama sama menggunakan hardware/software yang terhubung dengan jaringan. Tiap komputer, printer atau periferal yang terhubung dengan jaringan disebut **node**. Sebuah jaringan komputer dapat memiliki dua, puluhan, ribuan atau bahkan jutaan node.

Sebuah jaringan biasanya terdiri dari 2 atau lebih komputer yang saling berhubungan diantara satu dengan yang lain, dan saling berbagi sumber daya misalnya CDROM, Printer, pertukaran file, atau memungkinkan untuk saling berkomunikasi secara elektronik. Komputer yang terhubung tersebut, dimungkinkan berhubungan dengan media kabel, saluran telepon, gelombang radio, satelit, atau sinar infra merah.

Jenis-Jenis jaringan

Ada 3 macam jenis Jaringan/Network yaitu :

a. Local Area Network (LAN) /Jaringan Area Lokal.

Sebuah LAN, adalah jaringan yang dibatasi oleh area yang relatif kecil, umumnya dibatasi oleh area lingkungan seperti sebuah perkantoran di sebuah gedung, atau sebuah sekolah, dan biasanya tidak jauh dari sekitar 1 km persegi.

Beberapa model konfigurasi LAN, satu komputer biasanya dijadikan sebuah *file erver*. Yang mana digunakan untuk menyimpan perangkat lunak (*software*) yang mengatur aktifitas jaringan, ataupun sebagai perangkat lunak yang dapat digunakan oleh komputerkomputer yang terhubung ke dalam network. Komputer-komputer yang terhubung ke dalam jaringan (*network*) itu biasanya disebut dengan *workstation.* Biasanya kemampuan *workstation* lebih di bawah dari *file server* dan mempunyai aplikasi lain di dalam harddisknya selain aplikasi untuk jaringan. Kebanyakan LAN menggunakan media kabel untuk menghubungkan antara satu komputer dengan komputer lainnya.

b. Metropolitan Area Network (MAN) / Jaringan area Metropolitan

Sebuah MAN, biasanya meliputi area yang lebih besar dari LAN, misalnya antar wilayah dalam satu propinsi. Dalam hal ini jaringan menghubungkan beberapa buah jaringan-jaringan kecil ke dalam lingkungan area yang lebih besar, sebagai contoh yaitu : jaringan Bank dimana beberapa kantor cabang sebuah Bank di dalam sebuah kota besar dihubungkan antara satu dengan lainnya. Misalnya Bank BNI yang ada di seluruh wilayah Ujung Pandang atau Surabaya.

c. Wide Area Network (WAN) / Jaringan area Skala Besar

Wide Area Networks (WAN) adalah jaringan yang lingkupnya biasanya sudah menggunakan sarana Satelit ataupun kabel bawah laut sebagai contoh keseluruhan jaringan BANK BNI yang ada di Indonesia ataupun yang ada di Negara-negara lain. Menggunakan sarana WAN, Sebuah Bank yang ada di Bandung bisa menghubungi kantor cabangnya yang ada di Hongkong, hanya dalam beberapa menit. Biasanya WAN agak rumit dan sangat kompleks, menggunakan banyak sarana untuk menghubungkan antara LAN dan WAN ke dalam Komunikasi Global seperti Internet. Tapi bagaimanapun juga antara LAN, MAN dan WAN tidak banyak berbeda dalam beberapa hal, hanya lingkup areanya saja yang berbeda satu diantara yang lainnya.

Protokol

Protokol adalah aturan-aturan main yang mengatur komunikasi diantara beberapa komputer di dalam sebuah jaringan, aturan itu termasuk di dalamnya petunjuk yang berlaku bagi cara-cara atau metode mengakses sebuah jaringan, topologi fisik, tipe-tipe kabel dan kecepatan transfer data. Protokol-Protokol yang dikenal adalah sebagai berikut :

1. Ethernet
2. Local Talk
3. Token Ring
4. FDDI
5. ATM

Ethernet

Protocol Ethernet sejauh ini adalah yang paling banyak digunakan, Ethernet menggunakan metode akses yang disebut CSMA/CD *(Carrier Sense Multiple Access/Collision Detection)*. Sistem ini menjelaskan bahwa setiap komputer memperhatikan ke dalam kabel dari network sebelum mengirimkan sesuatu ke dalamnya. Jika dalam jaringan tidak ada aktifitas atau bersih komputer akan mentransmisikan data, jika ada transmisi lain di dalam kabel, komputer akan menunggu dan akan mencoba kembali transmisi jika jaringan telah bersih. kadangkala dua buah komputer melakukan transmisi pada saat yang sama, ketika hal ini terjadi, masing-masing komputer akan mundur dan akan menunggu kesempatan secara acak untuk mentransmisikan data kembali. metode ini dikenal dengan koalisi, dan tidak akan berpengaruh pada kecepatan transmisi dari network.

Protokol Ethernet dapat digunakan untuk pada model jaringan Garis lurus , Bintang, atau Pohon . Data dapat ditransmisikan melewati kabel twisted pair, koaksial, ataupun kabel fiber optic pada kecepatan 10 Mbps.

LocalTalk

LocalTalk adalah sebuh protokol network yang di kembangkan oleh Apple Computer, Inc. untuk mesin-mesin komputer Macintosh . Metode yang digunakan

oleh LocalTalk adalah CSMA/CA *(Carrier Sense Multiple Access with Collision Avoidance)*. Hampir sama dengan CSMA/CD.. Adapter LocalTalk dan cable twisted pair khusus dapat digunakan untuk menghubungkan beberapa computer melewati port serial. Sistem Operasi Macintosh memungkinkan koneksi secara jaringan peer-to-peer tanpa membutuhkan tambahan aplikasi khusus Protokol LocalTalk dapat digunakan untuk model jaringan Garis Lurus , Bintang , ataupun model Pohon dengan menggunakan kabel twisted pair . Kekurangan yang paling mencolok yaitu kecepatan transmisinya. Kecepatan transmisinya hanya 230 Kbps.

Token Ring

Protokol Token di kembangkan oleh IBM pada pertengahan tahun 1980. Metode Aksesnya melalui lewatnya sebuah token dalam sebuah lingkaran seperti Cincin . Dalam lingkaran token, komputer-komputer dihubungkan satu dengan yang lainnya seperti sebuah cincin. Sebuah Sinyal token bergerak berputar dalam sebuah lingkaran (cincin) dalam sebuah jaringan dan bergerak dari sebuah komputer-menuju ke komputer berikutnya, jika pada persinggahan di salah satu komputer ternyata ada data yang ingin ditransmisikan, token akan mengangkutnya ke tempat dimana data itu ingin ditujukan, token bergerak terus untuk saling mengkoneksikan diantara masing-masing komputer. protokol Token Ring membutuhkan model jaringan Bintang dengan menggunakan kabel twisted pair atau kabel fiber optic . Dan dapat melakukan kecepatan transmisi 4 Mbps atau 16 Mbps. Sejalan dengan perkembangan Ethernet, penggunaan Token Ring makin berkurang sampai sekarang.

FDDI

Fiber Distributed Data Interface (FDDI) adalah sebuah Protokol jaringan yang menghubungkan antara dua atau lebih jaringan bahkan pada jarak yang jauh . Metode aksesnyayang digunakan oleh FDDI adalah model token . FDDI menggunakan dua buah topologi ring secara fisik. Proses transmisi baiasanya menggunakan satu buah ring, namun jika ada masalah ditemukan akan secara otomatis menggunakan ring yang kedua. Sebuah keuntungan dari FDDI adalah kecepatan dengan menggunakan fiber optic cable pada kecepatan 100 Mbps.

ATM

ATM adalah singkatan dari *Asynchronous Transfer Mode* (ATM) yaitu sebuah protokol jaringan yang mentransmisikan pada kecepatan 155 Mbps atau lebih . ATM mentarnsmisikan data kedalam satu paket dimana pada protokol yang lain mentransfer pada besar-kecilnya paket. ATM mendukung variasi media seperti video, CD-audio, dan gambar. ATM bekerja pada model topologi Bintang , dengan menggunakan Kabel fiber optic ataupun kabel twisted pair . ATM pada umumnya digunakan untuk menghubungkan dua atau lebih LAN . dia juga banyak dipakai oleh *Internet Service Providers* (ISP) untuk meningkatkan kecepatan akses Internet untuk klien mereka.

Perangkat keras yang diperlukan

Perangkat keras yang dibutuhkan untuk membangun sebuah jaringan komputer yaitu : Komputer, Card Network, Hub, dan segala sesuatu yang berhubungan dengan koneksi jaringan seperti: Printer, CDROM, Scanner, Bridges, Router dan lainnya yang dibutuhkan untuk process transformasi data didalam jaringan, seperti:

1. File Servers
2. Workstations
3. Network Interface Cards
4. Concentrators/Hubs
5. Repeaters
6. Bridges
7. Routers

File Servers

Sebuah file server merupakan jantungnya kebayakan Jaringan, merupakan komputer yang sangat cepat, mempunyai memori yang besar, harddisk yang memiliki kapasitas besar, dengan kartu jaringan yang cepat. Sistem operasi jaringan tersimpan disini, juga termasuk didalamnya beberapa aplikasi dan data yang dibutuhkan untuk jaringan. Sebuah file server bertugas mengontrol komunikasi dan informasi diantara node/komponen dalam suatu jaringan. Sebagai contoh mengelola

pengiriman file database atau pengolah kata dari workstation atau salah satu node, ke node yang lain, atau menerima email pada saat yang bersamaan dengan tugas yang lain....terlihat bahwa tugas file server sangat kompleks, dia juga harus menyimpan informasi dan membaginya secara cepat. Sehingga minimal sebuah file server mempunyai beberpa karakter seperti tersebut di bawah ini :

- Processor minimal 166 megahertz atau processor yang lebih cepat lagi (Pentium Pro, Pentium II, PowerPC).
- Sebuah Harddisk yang cepat dan berkapasitas besar atau kurang lebih 10 GB
- Sebuah RAID (Redundant Array of Inexpensive Disks).
- Sebuah tape untuk back up data (contohnya . DAT, JAZ, Zip, atau CDRW)
- Mempunyai banyak port network
- Kartu jaringan yang cepat dan Reliabilitas
- Kurang lebih 32 MB memori

Workstations

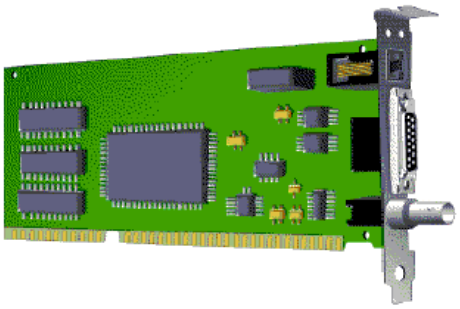
Keseluruhan komputer yang terhubung ke file server dalam jaringan disebut sebagai workstation. Sebuah workstation minimal mempunyai ; Kartu jaringan, Aplikasi jaringan (sofware jaringan), kabel untuk menghubungkan ke jaringan, biasanya sebuah workstation tidak begitu membutuhkan Floppy karena data yang ingin di simpan bisa dan dapat diletakkan di file server. Hampir semua jenis komputer dapat digunakan sebagai komputer workstation.

Network Interface Cards (NIC) atau Kartu Jaringan

Kartu Jaringan (NIC) merupakan perangkat yang menyediakan media untuk menghubungkan antara komputer, kebanyakan kartu jaringan adalah kartu inernal, yaitu kartu jaringan yang di pasang pada slot ekspansi di dalam komputer. Beberapa komputer seperti komputer MAC, menggunakan sebuah kotak khusus yang ditancapkan ke port serial atau SCSI port komputernya. Pada komputer notebook ada slot untuk kartu jaringan yang biasa disebut PCMCIA slot. Kartu jaringan yang banyak terpakai saat ini adalah : kartu jaringan Ethernet, LocalTalk konektor, dan kartu jaringan Token Ring. Yang saat ini populer digunakan adalah Ethernet, lalu diikuti oleh Token Ring, dan LocalTalk,

Ethernet Card / Kartu Jaringan Ethernet

Kartu jaringan Ethernet biasanya dibeli terpisah dengan komputer, kecuali seperti komputer Macintosh yang sudah mengikutkan kartu jaringan Ethernet didalamnya. kartu Jaringan ethernet umumnya telah menyediakan port koneksi untuk kabel Koaksial ataupun kabel twisted pair, jika didesain untuk kabel koaksial konenektorya adalah BNC, dan apabila didesain untuk kabel twisted pair maka akan punya konektor RJ-45. Beberapa kartu jaringan ethernet kadang juga punya konektor AUI. Semua itu di koneksikan dengan koaksial, twisted pair,ataupun dengan kabel fiber optik.



Gambar 2.21 Kartu Jaringan Ethernet Dari Atas Ke Bawah : konektor RJ-45, konektor AUI, dan konektor BNC

LocalTalk Connectors/Konektor LocalTalk

LocalTalk adalah kartu jaringan buat komputer macintosh, ini menggunakan sebuah kotak adapter khusus dan kabel yang terpasang ke Port untuk printer. Kekurangan dari LocalTalk dibandingkan Ethernet adalah kecepatan laju transfer datanya, Ethernet bi Jaringan komputer bukanlah sesuatu yang baru saat ini. Hampir di setiap perusahaan terdapat jaringan komputer untuk memperlancar arus informasi di dalam perudahaan tersebut. Internet yang mulai populer saat ini adalah suatu jaringan komputer raksasa yang merupakan jaringan jaringan komputer yang terhubungan dan dapat saling berinteraksi. Hal ini dapat terjadi karena adanya perkembangan teknologi jaringan yang sangat pesat, sehingga dalam beberapa tahun saja jumlah pengguna jaringan komputer yang tergabung dalam Internet berlipat ganda.asanya dapat sampai 10 Mbps, sedangkan LocalTalk hanya dapat beroperasi pada kecepatan 230 Kbps atau setara dengan 0.23 Mps

Token Ring Cards

Kartu jaringan Token Ring terlihat hampir sama dengan Kartu jaringan Ethernet. Satu perbedaannya adalah tipe konektor di belakang KArtu jaringannya, Token Ring umumnya mempunyai tipe konektor 9 Pin DIN yang menyambung Kartu jaringan ke Kabel Network.

Hub/Konsentrator

Sebuah Konsentrator/Hub adalah sebuah perangkat yang menyatukan kabel-kabel network dari tiap-tiap workstation, server atau perangkat lain. Dalam topologi Bintang, kabel twisted pair datang dari sebuah workstation masuk kedalam hub. Hub mempunyai banyak slot concentrator yang mana dapat dipasang menurut nomor port dari card yang dituju.

Ciri-ciri yang dimiliki Konsentrator adalah :

- Biasanya terdiri dari 8, 12, atau 24 port RJ-45
- Digunakan pada topologi Bintang/Star
- Biasanya di jual dengan aplikasi khusus yaitu aplikasi yang mengatur
- manjemen port tersebut.
- Biasanya disebut hub
- Biasanya di pasang pada rak khusus, yang didalamnya ada Bridges, router

Repeaters

Contoh yang paling mudah adalah pada sebuah LAN menggunakan topologi Bintang dengan menggunakan kabel unshielded twisted pair. Dimana diketahui panjang maksimal untuk sebuah kabel unshileded twisted pair adalah 100 meter, maka untuk menguatkan sinyal dari kabel tersebut dipasanglah sebuah repeater pada jaringan tersebut.

Bridges / Jembatan

Adalah sebuah perangkat yang membagi satu buah jaringan kedalam dua buah jaringan, ini digunakan untuk mendapatkan jaringan yang efisien, dimana kadang pertumbuhan network sangat cepat makanya di perlukan jembatan untuk itu. Kebanyakan Bridges dapat mengetahui masing-masing alamat dari tiap-tiap segmen

komputer pada jaringan sebelahnya dan juga pada jaringan yang lain di sebelahnya pula. Diibaratkan bahwa Bridges ini seperti polisi lalu lintas yang mengatur di persimpangan jalan pada saat jam-jam sibuk. Dia mengatur agar informasi di antara kedua sisi network tetap jalan dengan baik dan teratur. Bridges juga dapat di gunakan untuk mengkoneksi diantara network yang menggunakan tipe kabel yang berbeda ataupun topologi yang berbeda pula

Routers

Sebuah Router mengartikan informaari dari satu jaringan ke jaringan yang lain, dia hampir sama dengan Bridge namun agak pintar sedikit, router akan mencari jalur yang terbaik untuk mengirimkan sebuah pesan yang berdasakan atas alamat tujuan dan alamat asal. Sementara Bridges dapat mengetahui alamat masing-masing komputer di masing-masing sisi jaringan, router mengetahui alamat komputerr, bridges dan router lainnya. router dapat mengetahui keseluruhan jaringan melihat sisi mana yang paling sibuk dan dia bisa menarik data dari sisi yang sibuk tersebut sampai sisi tersebut bersih. Jika sebuah perusahaan mempunyai LAN dan menginginkan terkoneksi ke Internet, mereka harus membeli router. Ini berarti sebuah router dapat menterjemahkan informasi diantara LAN anda dan Internet. ini juga berarti mencarikan alternatif jalur yang terbaik untuk mengirimkan data melewati internet.

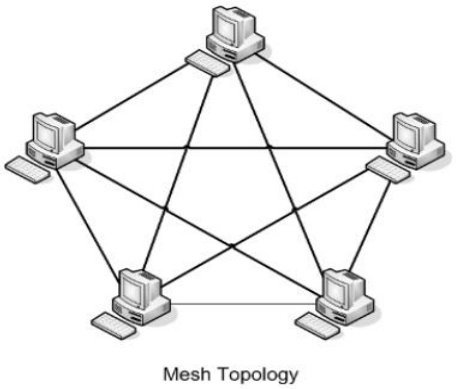
Ini berarti Router itu :

- Mengatur jalur sinyal secara effisien
- Mengatur Pesan diantara dua buah protocol
- Mengatur Pesan diantara topologi jaringan linear Bus dan Bintang(star)
- Mengatur Pesan diantara melewati Kabel Fiber optic, kabel koaaksialm atau
- kabel twisted pair

Topologi/Bentuk Jaringan

Topologi suatu jaringan didasarkan pada cara penghubung sejumlah nodeatau sentral dalam membentuk suatu sistem jaringan. Topologi jaringan yang umum dipakai adalah : Mess, Bintang (Star), Bus, Tree, dan Cincin (Ring).

a. Topologi Jaringan Mesh



Gambar 2.22 . Topologi Mesh.

Topologi mesh memiliki hubungan yang berlebihan antara peralatan-peralatanyang ada.Susunannya pada setiap peralatan yang ada didalam jaringan saling terhubung satu sama lain.jika jumlah peralatan yang terhubung sangat banyak, tentunya ini akan sangat sulit sekali untuk dikendalikandi bandingkan hanya sedikit peralatan saja yang terhubung.

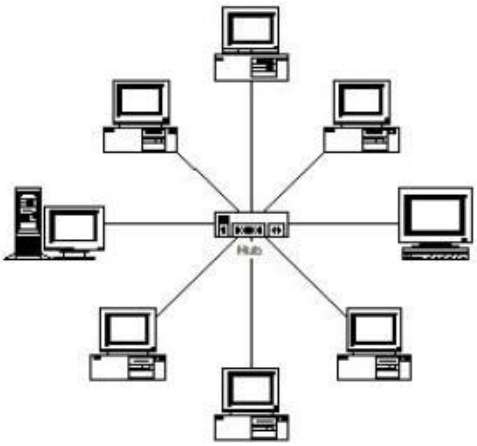
Keuntungan Topologi Mesh

- Keuntungan utama dari penggunaan topologi mesh adalah fault tolerance.
- Terjaminnya kapasitas channel komunikasi, karena memiliki hubungan yang berlebih.
- Relatif lebih mudah untuk dilakukan troubleshoot.

Kerugian Topologi Mesh

- Sulitnya pada saat melakukan instalasi dan melakukan konfigurasi ulang saat jumlah komputer dan peralatan-peralatan yang terhubung semakin meningkat jumlahnya.
- Biaya yang besar untuk memelihara hubungan yang berlebih.

b. Topologi Jaringan Bintang (Star)



Gambar 2.23 Topologi Star

Merupakan kontrol terpusat, semua link harus melewati pusat yang menyalurkan data tersebut kesemua simpul atau client yang dipilihnya. Simpul pusat dinamakan stasium primer atau server dan lainnya dinamakan stasiun sekunder atau client server. Setelah hubungan jaringan dimulai oleh server maka setiap client server sewaktu-waktu dapat menggunakan hubungan jaringan tersebut tanpa menunggu perintah dari server. Terdapat keuntungan dan kerugian dari tipe ini yaitu:

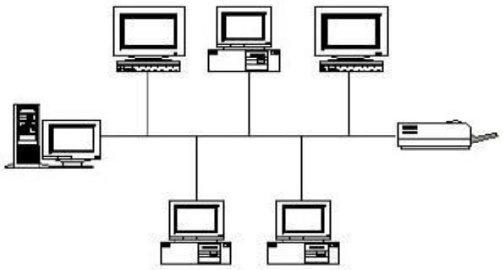
Keuntungan:

- Paling fleksibel.
- Pemasangan/perubahan stasiun sangat mudah dan tidak mengganggu bagian jaringan lain.
- Kontrol terpusat.
- Kemudahan deteksi dan isolasi kesalahan/kerusakan.
- Kemudahaan pengelolaan jaringan.

Kerugian:

- Boros kabel.
- Perlu penanganan khusus.
- Kontrol terpusat (HUB) jadi elemen kritis.

c. Topologi Jaringan Bus

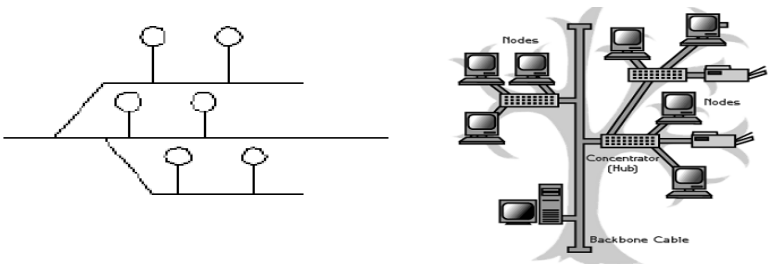


Gambar 2.24 Topologi BUS.

Topologi bus terlihat pada skema di atas. Terdapat keuntungan dan kerugian dari tipe ini yaitu:

Keuntungan:	Kerugian:
- Hemat kabel.	- Deteksi dan isolasi kesalahan sangat kecil.
- Layout kabel sederhana.	- Kepadatan lalu lintas.
- Mudah dikembangkan.	- Bila salah satu client rusak, maka jaringan tidak bisa berfungsi.
	- Diperlukan repeater untuk jarak jauh.

d. Topologi Jaringan Pohon (Tree)



Gambar 2.25 Topologi Tree.

Topologi jaringan ini disebut juga sebagai topologi jaringan bertingkat. Topologi ini biasanya digunakan untuk interkoneksi antar sentral dengan hirarki yang

berbeda. Untuk hirarki yang lebih rendah digambarkan pada lokasi yang rendah dan semakin keatas mempunyai hirarki semakin tinggi. Topologi jaringan jenis ini cocok digunakan pada sistem jaringan komputer.

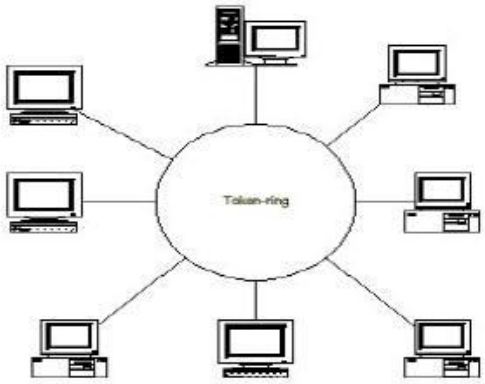
Kelebihan topologi tree

- Seperti topologi star perangkat terhubung pada pusat pengendali /HUB.
- Tetapi HUB dibagi menjadi dua,central HUB,dan secondary HUB.
- Topologi tree ini memiliki keunggulan lebih mampu menjangkau jarak yang lebih jauh dengan mengaktifkan fungsi Repeater yang dimiliki oleh HUB.

Kelemahan topologi tree

- kabel yang digunakan menjadi lebih banyak sehingga diperlukan perencanaan yang matang dalam pengaturannya , termasuk di dalamnya adalah tata letak ruangan.

e. Topologi Jaringan Cincin (Ring)



Gambar 2.26 Topologi Token RING.

Topologi TokenRING terlihat pada skema di atas. Metode token-ring (sering disebut ring saja) adalah cara menghubungkan komputer sehingga berbentuk ring (lingkaran). Setiap simpul mempunyai tingkatan yang sama. Jaringan akan disebut sebagai loop, data dikirimkan kesetiap simpul dan setiap informasi yang diterima simpul diperiksa alamatnya apakah data itu untuknya atau bukan. Terdapat

keuntungan dan kerugian dari tipe ini yaitu:

Keuntungan:	Kerugian:
- Hemat kabel.	- Peka kesalahan.
	- Pengembangan jaringan lebih kaku.

2.2.4 Internet

Pengertian

Istilah INTERNET berasal dari bahasa Latin inter, yang berarti *"antara"*. Secara kata per kata INTERNET berarti jaringan antara atau penghubung. Memang itulah fungsinya, INTERNET menghubungkan berbagai jaringan yang tidak saling bergantung pada satu sama lain sedemikian rupa, sehingga mereka dapat berkomunikasi. Sistem apa yang digunakan pada masing-masing jaringan tidak menjadi masalah, apakah sistem DOS atau UNIX.

Sementara jaringan lokal biasanya terdiri atas komputer sejenis (misalnya DOS atau UNIX), INTERNET mengatasi perbedaan berbagai sistem operasi dengan menggunakan "bahasa" yang sama oleh semua jaringan dalam pengiriman data. Pada dasarnya inilah yang menyebabkan besarnya dimensi INTERNET.

Dengan demikian, definisi INTERNET ialah "jaringannya jaringan", dengan menciptakan kemungkinan komunikasi antar jaringan di seluruh dunia tanpa bergantung kepada jenis komputernya.

Kesimpulan:

- Definisi INTERNET : Internet merupakan hubungan antar berbagai jenis komputer dan jaringan di dunia yang berbeda sistem operasi maupun aplikasinya di mana hubungan tersebut memanfaatkan kemajuan media komunikasi (telepon dan satelit) yang menggunakan protokol standar dalam berkomunikasi yaitu protokol TCP/IP.
- Fungsi : Internet merupakan media komunikasi dan informasi modern.

Sejarah Terbentuknya Internet

Banyak hal di INTERNET hanya dapat dimengerti dengan mengetahui latar-belakang perkembangannya.

ARPANet

Pada tahun 1969 ARPA (Advanced Research Project Agency), sebuah bagian dalam kementerian Pertahana Amerika Serikat memulai sebuah proyek, yang di satu sisi menciptakan jalur komunikasi yang tak dapat dihancurkan dan disisi lain memudahkan kerjasama antar badan riset diseluruh negeri, seperti juga industri senjata. Maka terbentuklah ARPANet.

Bila pada awalnya komputer sejenis yang melakukan pertukaran data, bertambahnya komputer dengan berbagai sistem operasi lain menuntut solusi baru komunikasi yang tak terbatas antar semua badan yang tergabung dalam jaringan.

Internetting Project

Untuk itu dibuat ***Internetting Project***, yang mengembangkan lebih lanjut hasil yang telah dicapai dalam ARPANet, agar media komunikasi baru ini juga dapat dimanfaatkan oleh berbagai sistem komputer yang tergabung. Kemudian vendor-vendor komputer meramaikan lalu lintas jaringa tersebut untuk berbagai kebutuhan sehingga terciptalah INTERNET.

Protokol Internet: TCP/IP

Seperti telah disebutkan di atas, INTERNET terbentuk dari jaringan-komputer yang tersebar di seluruh dunia. Masing-masing jaringan-komputer terdiri dari tipe-tipe komputer yang berbeda dengan jaringan yang lainnya. Maka diperlukan sebuah protokol yang mampu mengintegrasikan seluruh jaringan komputer tersebut.

Solusinya adalah sebuah protokol pengiriman data yang tak bergantung pada jenis komputer dan digunakan oleh semua komputer untuk saling bertukar data. Agar data tidak hanya dapat dikirim dan diterima, melainkan juga dapat dimanfaatkan oleh setiap komputer, diperlukan program standar yang mengolah data tersebut pada sistem yang berkaitan.

Protokol pengiriman merupakan sebuah konvensi (kesepakatan) yang

menetapkan dengan cara apa data dikirimkan dan bagaimana kesalahan yang terjadi dikenali serta dipecahkan. Secara sederhana prose pengiriman data terdiri atas dua langkah.

Pertama, data yang akan dikrimkan (misalnya sebuah file teks) dibagi ke dalam paket data berukuran data berukuran sama (*paket*), kemudian dikirimkan satu per satu. Di Internet, protokol ini disebut IP (*Internet Protocol*).

Kedua, harus dijamin setiap paket data sampai ke alamat yang benar dan semuanya benar diterima. Untuk itu diperlukan protokol lainnya, yaitu *Transmission Control Protocol* (TCP) mengaitkan sebuah blok data pada paket data IP, yang antara lain mengandung informasi mengenai alamat, jumlah total paket data dan urutan setiap paket yang membentuk paket tersebut. Hanya secara bersamaan kedua protokol membentuk kesatuan yang berfungsi, karena itu biasanya disebut TCP/IP.

Dengan adanya TCP/IP ini, INTERNET memiliki 3 keuntungan :

- Memberi kesempatan INTERNET menggunakan jalur komunikasi yang sama untuk pemakai yang berbeda pada saat yang sama. Karena paket-paket data tidak perlu dikirimkan bersama-sama, jalur komunikasi dapat membawa segala tipe paket data sementara mereka dikirimkan dari tempat yang satu ke tempat yang lain. Sebagai contoh, bayangkan sebuah jalan raya di mana mobil bergerak sepanjang jalan yang sama walaupun mereka menuju ke tempat-tempat yang berbeda-beda.
- Memberi INTERNET fleksibilitas. Sementara paket-paket data bergerak, mereka bergerak dari satu host ke host lain sampai mencapai tujuan akhir. Jika sebuah jalur komunikasi tidak berfungsi, sistem yang mengontrol aliran data dapat menggunakan jalur alternatif. Maka, paket-paket data dapat bergerak melalui jalur-jalur yang berbeda-beda.
- Meningkatkan kecepatan transmisi data. Sebagai contoh, jika terjadi kesalahan, TCP meminta host asal mengirm kembali hanya paket-paket data yang mengandung kesalahan, bukan semua paket data. Ini berarti meningkatkan kecepatan transmisi data.

Cara Akses ke Internet

1. Sambungan langsung ke Network

   Anda dapat menggunakan sebuah komputer yang secara langsung mempunyai hubungan ke INTERNET. Sebagai contoh, Anda mungkin menggunakan sebuah PC yang merupakan bagian dari sebuah jaringan komputer yang mempunyai hubungan ke INTERNET. Dalam kasus ini, sistem Anda menjadi host INTERNET penuh, yaitu mempunyai alamat elektronik tersendiri.

2. Sambungan dengan menggunakan SLIP/PPP

   Untuk menggunakan hubungan dial-up telepon, Anda memerlukan sebuah alat untuk mengkonversi sinyal komputer (digital) menjadi sinyal telepon (analog), dan sebaliknya. Alat untuk mengkonversi sinyal digital ke sinyal analog disebut *modulator*. Sedang, alat untuk mengkonversikan sinyal analog ke sinyal digital disebut *demodulator*. Untuk mengakses ke INTERNET melalui hubungan telepon, Anda memerlukan sebuah modem (*modulator-demodulator*). Selain itu diperlukan juga TCP/IP dan software SLIP atau PPP seperti Linux, Warp, dll.

3. Sambungan langsung ke On-line Service seperti BBS, Compuserve.

   Untuk menjadi sebuah host INTERNET tanpa harus memiliki hubungan full-time ke INTERNET (yang umumnya sangat mahal), ada sebuah cara mensetup sebuah host INTERNET melalui hubungan telepon. Untuk melakukan hal tersebut, Anda perlu mengadakan perjanjian dengan sebuah host INTERNET yang lain yang bertindak sebagai titik hubungan. Selanjutnya, diperlukan sejumlah program yang disebut sebagai PPP (Point to Point Protocol) dan SLIP (Serial Line Internet Protocol) dalam workstation. Setelah workstation menghubungi host INTERNET melalui jalur telepon, PPP menyediakan kemampuan TCP/IP untuk workstation tersebut.

Domain Name System

Tentunya juga harus ada sebuah metode yang memungkinkan pengiriman kepada masing-masing komputer di INTERNET. Pada awalnya disusun sebuah daftar yang memuat nama semua jaringan yang tergabung dalam INTERNET.

Dengan tingkat pertumbuhan INTERNET yang sangat tinggi tidak mungkin lagi menjaga daftar-daftar tetap aktual. Karena Internet menggunakan sebuah sistem alamat yang dapat diolah oleh komputer khusus (disebut *Router*), yang terdapat antara masing-masing jaringan.

Alamat IP

Sebuah alamat IP terdiri atas sebuah angka biner 32-bit, yang menggambarkan lokasi jaringan hingga komputer dalam jaringan tersebut yang harus dicapai. Dari sanalah Router memilih jalur yang paling menguntungkan. Artinya, Internet menentukan sendiri jalan "melalui" banyak jaringan yang tergabung antara dua tempat, sehingga hampir tak mungkin merusak media komunikasi ini. Bila misalnya sebuah kabel penghubung rusak oleh pekerjaan galian tanah, INTERNET mengalihkan pengiriman paket ke jalur lain. Hal ini disebut *Dynamic Rerouting*.

Alamat DNS

Karena angka biner tidak mudah diingat, maka dikembangkan sistem *Domain Name System* (DNS). Disini alamat disusun dalam sebuah hierarki berbagai wilayah (domain = wilayah), yang mewakili sebuah kelompok *host*
tertentu. Host adalah komputer dalam jaringan lokal (LAN) atau Wide Area Network (WAN), yang diakses oleh komputer lain dalam jaringan tersebut.

Agar penyampaian otomatis oleh Router tetap berfungsi, angka biner tetap digunakan. Bila Anda memberikan sebuah alamat DNS, pertama-tama data dikirimkan ke sebuah *Server* dan diubah menjadi alamat Ip yang dapat dibaca oleh Komputer.

Contoh berikut ini akan menjelaskan sistem alamat DNS. Seperti pada alamat IP, alamat tersebut juga mengandung informasi yang dibutuhkan untuk identifikasi komputer yang diingikan. Setiap bagian alamat DNS harus dipisahkan dengan sebuah titik.

Gopher.bppt.id

Gopher adalah nama sebuah komputer dalam jaringan "bppt" dalam domain "id", yang berarti Indonesia. Seperti Anda lihat pada contoh di atas, mula-mula ditulis nama host yang diinginkan, disusul oleh bagian-bagian yang semakin besar hingga tingkat tertinggi, yaitu domain.

Untuk mencapai komputer tersebut, alamat dibaca dari kanan ke kiri. Mula-mula jaringan tempat Anda mengakses INTERNET menciptakan hubungan dengan Router tertinggi yang mengelola semua alamat dalam domain "id".Domain diatas mewakili sebuah wilayah geografis, singkatan "id" berarti Indonesia. Selain itu terdapat pula berbagai domain tematik, yang dikenali pada singakatan paling kanan dalam alamat DNS.

Tabel 2.35 Tabel domain tematik

Singkatan	**Jenis Domain**
com	perusahaan komersial
edu	badan pendidikan (misalnya universitas atau institut)
gov	lembaga pemerintahan non-militer
mil	militer
net	jaringan
org	organisasi lainnya
int	organisasi internasional
ac	badan pendidikan

Tabel 2.36 Domain geografis

Singkatan	**Artinya**	**Singkatan**	**Artinya**	**Singkatan**	**Artinya**
at	Austria	Pl	Polandia	de	Jerman
au	Australia	Pt	Portugal	be	Belgia
ch	Swiss	Se	Swedia	bg	Bulgaria
dk	Denmark	Uk	Inggris	br	Brazil

ee	Eslandia	Us	Amerika	fi	Finlandia
es	Spanyol	Cl	Cile	fr	Prancis
in	India	Ir	Irlandia	jp	Jepang

Dengan demikian alamat INTERNET standar adalah : username@domain

Bagaimana Mengakses Internet

1. Hardware

   Memiliki komputer dengan spesifikasi minimum sebagai berikut :

   - PC minimal 386 (486 DX-4 *recomended*)
   - Monitor Super VGA
   - RAM 4 MB (8 MB *recomended*, tergantung program aplikasi)
   - Memiliki Communication Port (RS-232)
   - Modem dengan kecepatan 14400 BPS atau lebih
   - Mouse
   - Multi Media (tergantung aplikasi pelayanan INTERNET)
   - Mampu menjalankan system Windows

2. Berlangganan dengan ISP (INTERNET Service Provider) atau Online Service
3. Software/Program Aplikasi (tergantung fasilitas INTERNET yang akan digunakan)

   - Eudora Light
   - Netscape
   - Internet Explorer
   - Internet Phone
   - Net 2 Phone
   - Net Meeting

Pelayanan Utama Dalam Internet

Software yang mendukung Internet menyediakan banyak pelayanan teknis. Bagian ini akan membahas empat pelayanan INTERNET yang paling penting dan mendasar.

1. *Pelayanan Mail*
   a. Pelayanan untuk mengirim dan menerima pesan-pesan. Setiap pesan yang dikirim dari satu sistem ke sistem-sistem yang lain menuju tujuan akhir. Di belakang layar, pelayanan

   b. mail memastikan bahwa pesan-pesan yang dikirim dan diterima secara lengkap pada alamat yang benar.

2. *Pelayanan telnet*
   a. Pelayanan yang memberikan kesempatan Anda menghubungi sistem remote atau sistem yang terletak di tempat yang jauh. Sebagai contoh, Anda dapat menggunakan telnet untuk menghubungi sebuah host/provider di negara lain. Setelah Anda menghubungi host tersebut, Anda dapat login ke host tersebut (username dan password yang sah). Setelah itu Anda dapat bertukar data melalui INTERNET.

3. *Pelayanan File Transfer Protocol (FTP)*
   a. Pelayanan transfer file dari satu sistem ke sistem lain. Dalam INTERNET, Anda dapat mentransfer file dari host ke sebuah host remote. Proses ini disebut *uploading,* dan sebaliknya jika Anda mentransfer file dari host remote ke host lokal disebut *downloading*.

4. *Pelayanan Client/Server*

   a. Program Client meminta dukungan program server. Sebagai contoh, Gopher client menampilkan menu, setelah Anda memilih perintah-perintah dalam menu, Gopher client menghubungi Gopher server yang sesuai (tidak perduli dimana lokasinya dalam INTERNET) dan mengeksekusi permintaan Anda.

Jenis – jenis Layanan Internet

- **Electronic Mail (E-mail)**

Fungsi : mengirim atau menerima surat ke/dari seluruh penjuru dunia.

Sebagai pemakai INTERNET, Anda dapat mengirim dan menerima pesan

dari pemakai INTERNET lain dari berbagai penjuru dunia. Namum selain pesan-pesan pribadi, dengan E-mail dapat juga mengirim dan menerima file binary. Maka secara virtual Anda dapat mengirim dan menerima segala tipe data. Sistem mail INTERNET adalah tulang punggung (dan motivasi awal) dari INTERNET itu sendiri.

Untuk dapat menerima surat elektronik, Anda harus memiliki kotak pos (mailbox) untuk menampung surat-surat yang masuk sebelum Anda sempat menbacanya. Sebuah kotak pos elektronik (*electronic mailbox*) sama dengan kotak pos dikantor pos. Siapapun bisa mengirim surat ke kotak pos, tetapi hanya pemiliknya yang bisa meneliti dan membuang isi kotak surat tersebut.

Alamat kotak pos untuk surat elektronik disebut E-mail address. Sebagai contoh :

budi @indo.net.id

budi : nama user

indo : nama provider

net : network/provider

id : nama domain geografis, yaitu Indonesia

Surat menyurat di INTERNET dilakukan dengan menggunakan program surat elektronik. Cara penggunaannya sangat mudah bila program dijalankan dibawah *user interface* grafis seperti Windows. Saat pengiriman hanya perlu diisikan alamat penerima dan subjek (topik) surat, kemudian isi surat langsung dapat diketik untuk dikirim. Jika perlu Anda dapat menyertakan file tertentu untuk dikirim bersama-sama dengan surat tersebut (*Attachments*).

Program surat elektronik yang popular digunakan adalah Eudora Mail. Eudora Mail menggunakan protokol yang disebut POP (*Post Office Protocol*) dan dibuat oleh Qualcomm Inc.

- **File Tranfer Protocol (FTP)**

  Fungsi :mengirim dan menerima file antar host dari seluruh penjuru dunia.

  Anonymous FTP memungkinkan pengaksesan ke server FTP dengan login anonymous tanpa memerlukan password. Anonymous FTP adalah salah satu dari pelayanan dalam Internet yang cukup penting. Dengan akses ke berbagai

anonymous FTP, Anda dapat memperoleh file-file secara grafis. Anda dapat menemukan program-program, gambar-gambar, majalah elektronik, artikel-artikel dalam kelompok diskusi tertentu. Salah satu program FTP adalah WS_FTP.

- **Tele Networking (TelNet)**

  Fungsi : mengakses komputer (host/server) dari jauh/Remote login.

  Telnet adalah program yang memungkinkan komputer kita menjadi terminal dari komputer lain di INTERNET. Telnet memungkinkan kita untuk masuk (*log in*) sebagai pemakai komputer jarak jauh dan menjalankan program komputer layanan yang ada dikomputer tersebut.

- **User's Network (UseNet)**

  UseNet adalah sistem kelompok diskusi di mana artikel-artikel didistribusikan ke seluruh dunia. UseNet memiliki ribuan kelompok diskusi, sehingga tidak heran jika UseNet meliputi segala macam topik yang mungkin Anda inginkan.

- **World Wide Web (WWW)**

  Sering disebut "the WEB"/"W3", merupakan sistem dalam internet yang memiliki fasilitas pencarian dan pemberian informasi yang cepat dengan menggunakan teknologi *hypertext*.

  Sebutan World Wide Web (Web=jaring laba-laba) sangat tepat untuk menggambarkan struktur data pada jaringan INTERNET. Berbeda dengan misalnya susunan data logis berstruktur pohon yang dikenal dari DOS. WWW memungkinkan penanganan atau akses yang jauh lebih fleksibel pada file yang dikelola.

  Di WWW, struktur sumber daya-INTERNET dapat dibandingkan dengan jaring laba-laba. Bila dilihat polanya, jaringan ini terdiri atas lingkaran-lingkaran berbagai ukuran yang berpusat pada titik tengah yang sama. Dari titik tengah ini terbentuk garis-garis penghubung yang tegak lurus pada lingkaran, sehingga terdapat titik simpul. Bila pada struktur pohon percabangan merupakan jalur hubungan, pada Web semua garis merupakan penghubung setiap titik simpul yang mengandung data. Pemilihan disini dilakukan dengan item Hypertext. Pada

titik simpul bisa terdapat sebuah komputer di Internet atau sebuah petunjuk untuk file tertentu pada sebuah komputer. Hal ini berarti, dengan memilih sebuah item Hypertext diciptakan hubungan dengan sebuah komputer pada suatu tempat di dunia, dimana Anda dapat melanjutkan perjalanan atau langsung ke sebuah file tertentu.

"Bahasa" World Wide Web:HTML

Untuk membuat Hypertext, dikembangkan sebuah bahasa pemrograman khusus yang memungkinkan pengikatan alamat WWW atau file dalam sebuah dokumen. Sesuai dengan fungsinya, bahasa pemrograman ini disebut *Hypertext Mark up Language* (HTML). File ini biasanya berextention ***.html**. Agar file yang berisi Hypertext ini bisa dikirimkan, diperlukan protokol pengiriman data yang spesifik yang disebut HyperText Transfer Protocol (HTTP).

Untuk menemukan setiap hubungan Hypertext digunakan Uniform Resource Locator (URL). Karena itu, halaman WWW juga disebut dokumen URL. Beberapa alamat browser/search engine di INTERNET :

Tabel 2.37 alamat browser/search engine di INTERNET

Nama	**Alamat elektronik**
Yahoo!	http://www.yahoo.com/
Alta Vista	http://www.altavista.digital.com/
Lycos	http://www.lycos.com/
WebCrawler	http://www.webcrawler.com/
Inktomi	http://www.cs.berkeley.edu/
DejaNews	http://www.dejanews.com/
BimaSakti	http://www.cs.utexas.edu/users/adison/cgi/bimasakti/

- **Internet Relay Chat (IRC)**

  Internet Relay Chat/IRC merupakan fasilitas untuk komunikasi langsung dengan

menggunakan keyboard. Anda dapat ambil bagian dalam komunikasi publik dengan sekelompok orang. Atau, jika Anda inginkan, Anda dapat menggunakan IRC untuk mengatur komunikasi pribadi dengan orang-orang tertentu, yaitu sejenis teleconference.

- **Internet Phone/Conference**

  Fasilitas untuk melakukan percakapan jarak jauh via INTERNET. Untuk itu diperlukan aplikasi khusus dan dukungan hardware multi media.

- **WAIS Server**

  WAIS (Wide Area Information Service) menyediakan cara lain untuk menemukan informasi yang tersebar dalam INTERNET. WAIS mampu mengakses segala database yang besar (seperti dokumen, file berisi gambar, video dan suara).

- **Gopher**

  Internet menyediakan banyak informasi yang dapat diakses penggunanya lewat sistem menu. Seorang pengguna INTERNET dihadapkan pada sebuah menu yang bercabang-cabang. Untuk menuju ke informasi atau data yang dituju, seorang pengguna menyeleksi pilihan-pilihan yang disediakan hingga masuk ke topik yang diinginkan. Fasilitas demikian disebut Gopher.

- **Mailing List**

  Kelompok diskusi - fasilitas ini dibangun menggunakan teknik yang sama dengan proses penyebaran surat elektronik. Dengan menggunakan fasilitas ini, sebuah berita/file dapat didistribusikan ke banyak pengguna sekaligus. Bahkan penggunanya dapat melakukan diskusi, seminar, ceramah, konferensi secara elektronik tanpa terikat dimensi ruang dan waktu. Diskusi dapat berlangsung setiap hari tanpa henti. Hasil yang diperoleh akan jauh lebih efektif daripada penyelenggaraan seminar/konferensi konvesional.

Tabel 2.38 Daftar alamat Mailing List dengan topik ekonomi

Mailing List	Keterangan
Nacubo%ctstatey.bitnet@vm1.nodak.edu	Bisnis,Keuangan, dan Administrasi Pro
Econ-dev@csn.org	Perkembangan ekonomi Internasional
economy@tecmetyvm.mty.itesm.mx	Ekonomi di negara-negara kurang berkembang
gc-INTERNET@uriacc.uri.edu	Ekonomi global
pol-econ@shsu.bitnet	Ekonomi politik

BEBERAPA ISTILAH INTERNET

American OnLine (AOL)

Layanan Komersial yang menyediakan browser Internet-nya sendiri.

Browser

Aplikasi yang ada di komputer Anda, mirip dengan aplikasi pengolah kata atau aplikasi lain, yang dirancang untuk membaca tipe file data tertentu. Untuk mengakses WWW, Anda membutuhkan program software internet browser ini. Browsing-metode untuk menjelajahi WWW Internet. Mem-*browse* WWW sering disebut sebagai *surfing the Net*.

CU-SeeMe

Aplikasi yang memungkinkan konferensi video pada Internet.

Cyberspace

Tempat virtual di mana orang dapat berkomunikasi dan bertemu.

Domain name

Bagian yang membentuk IP *address* pada Internet. Domain name terdiri dari dua bagian atau lebih yang terpisah oleh tanda titik. Bagian paling kiri adalah bagian yang paling penting, menunjukkan tujuannya. Contoh : WWW menunjukkan Web server mail. Sembarang *domain name* akan "melebur" menjadi hanya satu IP address yang unik untuk setiap mesin. Bagian sebelah kanan menunjukkan tipe *site*. Contoh :.gov menunjukkan negara asal site, misalnya id menunjukkan negara Indonesia.

E-Mail

Mirip surat pos, tetapi dikirimkan secara elektronik, sehingga tiba ditempat tujuan nyaris dalam tempo seketika itu juga. E-mail yang berisikan pesan dikirim ke komputer Anda dan akan terus ada di komputer itu sampai Anda mengambilnya (membuka dan menyimpannya atau membuangnya).

Eudora

Aplikasi *e-mail* desktop yang populer karena lingkungan grafis dan kemampuannya untuk bekerja di hampir semua jenis server (PC maupun Macintosh).

FAQ

Frequently Asked Questions - daftar yang memuat jawaban dari pertanyaan-pertanyaan yang paling sering diajukan tentang topik tertentu.

FTP (File Transfer Protocol)

Program yang digunakan pada komputer berbasis Windows untuk mentransfer file (software atau dokumen) pada Internet.

GIF (Graphics Interchange Format)

Jenis image yang seringkali dipakai pada *web pages* karena kemampuannya untuk me-*render* warna latar belakang sebagai transparan.

Gopher

Program berbasis teks yang digunakan untuk mem-*browser* daftar menu Internet pada berbagai server gopher.

Home Page

Halaman-halaman yang menampung berbagai informasi dari suatu perusahaan, institusi atau pribadi. Contoh White House home page berisikan informasi politik dan kebijakan Presiden AS.

HTML

HyperText Markup Language - bahasa standar yang digunakan *browser* Internet untuk membuat halaman dan dokumen yang dipajang pada Web. Selain memungkinkan komputer berkomunikasi, HTML juga menyediakan *link* di antara file-file yang ada di komputer yang berbeda dan dipisahkan oleh jarak yang jauh.

HTTP

HyperText Transfer Protocol - protokol yang memungkinkan bermacam-macam komputer yang saling berkomunikasi dengan menggunakan bahasa HTML.

HyperLink

Bagian dari Web page yang menghubungkan Anda ke file lain, menyediakan *seamless link* (hubungan) ke file-file pada komputer lain. Biasa dikenal sebagai *link*. Mengklik *hyperlink* berarti mem-*browse* WWW dan mengunjungi berbagai lokasi.

Hypermedia document

File data berisi banyak informasi yang dikirimkan melalui Internet ke komputer Anda dan dimunculkan secara grafis dalam cara yang *user friendly*.

Hypertext

Salah satu cara untuk me-link banyak file. Teks yang mengandung hypertext biasanya digarisbawahi. Untuk melihat ‘kait’nya Anda cukup mengklik pada teks

tersebut.

Internet

Jaringan komputer dunia yang menghubungkan jaringan-jaringan komputer komersiah regional di seluruh dunia.

Intranet.

Pemanfaatan Internet dalam LAN untuk menghubungkan oran-orang dalam satu perusahaan tanpa berhubungan dengan dunia luar.

IP address

Alamat IP yang terbentuk dari empat angka yang dipisahkan oleh tanda titik. Alamat yang bersifat unik ini akan diberikan setelah Anda mengirimkan e-mail ke Hostmaster@INTERNIC.NET.

ISP (Internet Service Provider)

Perusahaan penyedia jasa koneksi ke Internet.

Java

Bahasa pemrograman berbasis-obyek yang sederhana, dan tidak tergantung pada platform dan sistem operasi. Bahasa ini dikembangkan oleh Sun Microsystems dan digunakan untuk membuat HotJava.

JPEG (Joint Photographic Expert Group)

Image grafis dalam bentuk terkompresi yang sering digunakan pada Web pages.

Listserv

Komputer yang mengotomatiskan pengiriman e-mail (bertindak sebagai pusat distribusi untuk pesan-pesan e-mail). Salah satu contoh adalah EDTECH yang para anggotanya mendiskusikan topik-topik yang terkait dengan teknologi pendidikan.

Lycos

Salah satu *search engine*. Lycos dikembangkan oleh Carnegie Mellon University dan melakukan pencarian berdasarkan judul dan isi. Pembaruan indeks dilakukan setiap minggu, sedangkan pencariannya bisa dilakukan setiap hari.

MPEG (Moving Picture Experts Group)

Algoritma terkompresi untuk file video dan audio. Bentuk ini sering digunakan pada web.

Net

Singkatan dari cyberspace.

Netscape

Browser Internet yang juga merangkap fungsi e-mail, FTP, Telnet, dan Gopher.

Newsgroup

Salah satu cara untuk berkomunikasi dengan penjelajah Internet lainnnya.

Online

Sering digunakan untuk merujuk saat orang ber-Internet.

2.2.5 Model Proses Perangkat Lunak

Pendahuluan

Di dalam suatu industri dikenal berbagai macam proses, demikian juga halnya dengan industri perangkat lunak. Perbedaan proses yang digunakan akan menguraikan aktivitas-aktivitas proses dalam cara-cara yang berlainan. Perusahaan yang berbeda menggunakan proses yang berbeda untuk menghasilkan produk yang sama. Tipe produk yang berbeda mungkin dihasilkan oleh sebuah perusahaan dengan menggunakan proses yang berbeda. Namun beberapa proses lebih cocok dari lainnya untuk beberapa tipe aplikasi. Jika proses yang salah digunakan akan mengurangi

kualitas kegunaan produk yang dikembangkan. Karena banyaknya variasi dalam model proses yang digunakan maka tidak mungkin menghasilkan gambaran-gambaran yang reliabel untuk alokasi biaya dalam aktivitas-aktivitas ini.

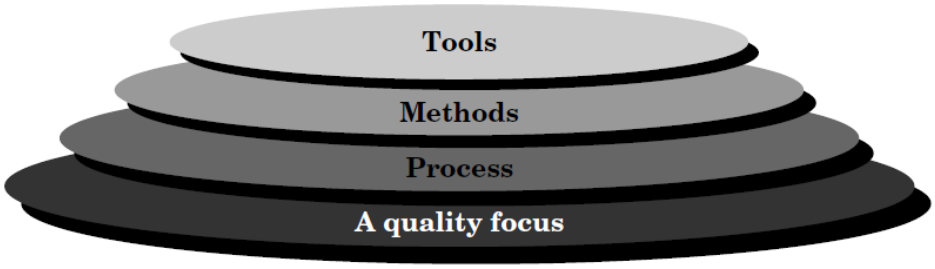
Proses perangkat lunak dapat didefinisikan sebagai sebuah kerangka kerja untuk tugas-tugas yang dibutuhkan dalam rangka membangun perangkat lunak dengan kualitas yang tinggi. Proses perangkat lunak menentukan pendekatan yang digunakan ketika perangkat lunak dikembangkan, tetapi pengembangan perangkat lunak juga meliputi teknologi yang mempopulasikan proses, metode teknis, serta alat-alat otomatis.

Modifikasi perangkat lunak biasanya lebih dari 60% dari total biaya pembuatan perangkat lunak. Presentase ini terus bertambah karena lebih banyak perangkat lunak dihasilkan dan dipelihara. Pembuatan perangkat lunak untuk suata perubahan adalah penting. Proses perangkat lunak komplek dan melibatkan banyak aktivitas. Seperti produk, proses juga memiliki atribut dan karakteristik seperti :

a. *Understandability*, yaitu sejauh mana proses secara eksplisit ditentukan dan bagaimana kemudahan definisi proses itu dimengerti.
b. *Visibility*, apakah aktivitas-aktivitas proses mencapai titik akhir dalam hasil yang jelas sehingga kemajuan dari proses tersebut dapat terlihat nyata/jelas.
c. *Supportability*, yaitu sejauh mana aktivitas proses dapat didukung oleh CASE.
d. *Acceptability*, apakah proses yang telah ditentukan oleh insinyur dapat diterima dan digunakan dan mampu bertanggung jawab selama pembuatan produk perangkat lunak.
e. *Reliability*, apakah proses didesain sedikian rupa sehingga kesalahan proses dapat dihindari sebelum terjadi kesalahan pada produk.
f. *Robustness*, dapatkah proses terus berjalan walaupun terjadi masalah yang tak diduga.
g. *Maintainability*, dapatkah proses berkembang untuk mengikuti kebutuhan atau perbaikan.
h. *Rapidity*, bagaimana kecepatan proses pengiriman sistem dapat secara lengkap memenuhi spesifikasi.

Proses Metode dan Alat Bantu

Rekayasa perangkat lunak merupakan sebuah teknologi yang terdiri atas beberapa lapisan. Banyak pendekatan keteknian yang harus berada pada sebuah komitmen dasar menuju kualitas. Manajemen kualitas total serta filosofinya mengangkat budaya pengembangan proses yang terus menerus, dan budaya itu sendiri membawa kepada perkembangan pendekatan yang semakin matang. Batu landasan yang menopang rekayasa perangkat lunak merupakan fokus kepada kualitas.



Gambar 2.27 Lapisan Rekayasa Perangkat Lunak

Fondasi untuk rekayasa perangkat lunak merupakan bentangan proses, yaitu perekat yang menjaga bentangan-bentangan teknologi secara bersama-sama dan memungkinkan perkembangan perangkat lunak komputer yang tepat waktu dan rasional. Berdasarkan gambar lapisan rekayasa perangkat lunak diatas dapat dijelaskan bahwa untuk menghasilkan perangkat lunak yang berkualitas diperlukan tools atau alat, methods atau teknik serta proses pembuatan yang berfokus pada kualitas.

Usaha yang berhubungan dengan rekayasa perangkat lunak dapat dikatagorikan kedalam tiga fase umum dengan tanpa mempedulikan area aplikasi, ukuran proyek, atau kompleksitasnya.

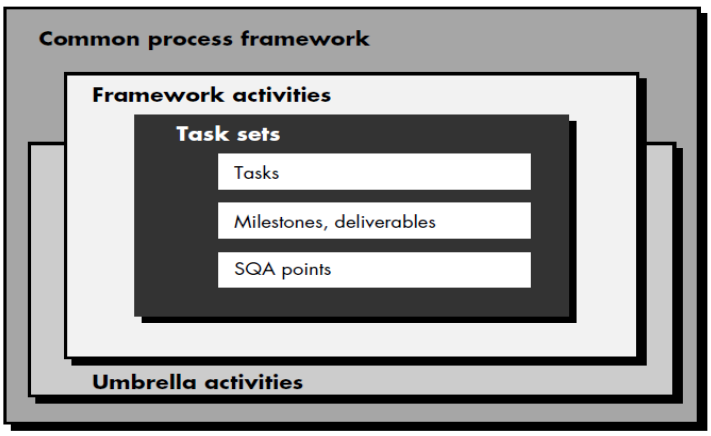
a. Fase Definisi (*Definition Phase*), berfokus pada “apa” (what); dimana pada definisi ini pengembangan perangkat lunak harus mengidentifikasi informasi apa yang akan diproses, fungsi dan unjuk kerja, apa yhang dibutuhkan, tingkah laku system seperti apa yang diharapkan, interface apa yang akan

dibangun, batasan desain apa yang ada, dan criteria validasi apa yang dibutuhkan untuk mendefinisikan system yang sukses.

b. Fase Pengembangan (*Development Phase*), berfokus pada how "bagaimana", yaitu dimana selama masa pengembangan perangkat lunak, teknisi harus mendefinisikan bagaimana data dikunstruksikan, bagaimana fungsi-fungsi diimplementasikan sebagai sebuah arsitektur perangkat lunak, bagaimana detail prosedur akan diimplementasikan, bagaimana interface ditandai, bagaimana rancangan akan diterjemahkan kedalam bahasa pemrograman, serta bagaimana pengujian akan dilakukan.
c. Fase Pemeliharaan (*Maintenance Phase*), berfokus pada perubahan "change", yang dihubungkan dengan koreksi kesalahan, penyesuaian yang dibutuhkan ketika lingkungan perangkat lunak berkembang, serta perubahan sehubungan dengan perkembangan yang disebabkan oleh perubahan pelanggan.

Proses Perangkat Lunak

Sebuah kerangka kerja proses umum dibangun dengan mendefinisikan sejumlah kecil aktivitas kerangka kerja yang bias diaplikasikan ke semua proyek perangkat lunak, tanpa melihat ukuran atau kompleksitasnya.



Gambar 2.28 Proses perangkat lunak

Pendekatan SEI memberikan sebuah pengukuran terhadap efektivitas global dari sebuah praktek perekayasaan perangkat lunak perusahaan dan membangun lima

tingkat kematangan proses, yang didefinisikan dengan cara berikut:

- Level 1 : Initial – proses perangkat lunak yang ditandai dengan ad hoc, dan bahkan kadang-kdang bersifat kacau.
- Level 2 : Repeatable – proses-proses manajemen proyek dasar dibangun untuk menelusuri masalah biaya, jadwal dan fungsionalitas.
- Level 3 : Defined – Proses perangkat lunak baik untuk aktivitas manajemen atau perekayasaan didokumentasikan, distandarkan, dan diintegrasikan kedalam proses perangkat lunak organisasi besar.
- Level 4 : Managed – Pengukuran detail terhadap proses perangkat lunak dan kualitas produksi dikumpulkan. Produk dan proses perangkat lunak dipahami secara kuantitatif dan dikontrol dengan menggunakan pengukuran secara detail.
- Level 5 : Optimizing – pertambahan proses yang terus menerus dimungkinkan oleh umpan balik kuantitatif dari proses dan dari gagasan inovatif pengujian serta teknologi.

Aktivitas pada Proses Perangkat Lunak

Serangkaian kegiatan dan hasil-hasil relevannya yang menghasilkan perangkat lunak sebagian besar dilakukan oleh pe-rekayasa perangkat lunak. Ada empat kegiatan/aktivitas pada proses perangkat lunak :

a. Spesifikikasi, Fungsionalitas perangkat lunak dan batasan kemampuan operasinya harus didefinisikan.
b. Pengembangan, Perangkat lunak yang memenuhi spesifikasi harus di produksi.
c. Validasi, Perangkat lunak harus divalidasi untuk menjamin bahwa perangkat lunak melakukan apa yang diinginkan oleh pelanggan.
d. Evolusi, Perangkat lunak harus berkembang untuk memenuhi kebutuhan pelanggan.

Model Proses Perangkat Lunak

Tidak mungkin untuk mengoptimalkan semua atribut proses secara serentak. Contohnya, jika pengembangkan proses cepat dilakukan mungkin kita perlu mengurangi visibility proses karena pembuatan proses yang nyata berarti pembuatan dokumen secara teratur. Ini akan memperlambat proses. Model proses perangkat lunak masih menjadi object penelitian, tapi sekarang ada banyak model umum atau paradigma.

Model proses perangkat lunak bisa mencakup kegiatan yang merupakan bagian dari proses perangkat lunak, produk perangkat lunak, dan peran orang yang terlibat pada rekayasa perangkat lunak (Perekayasa Perangkat Lunak). Contoh jenis model proses perangkat lunak

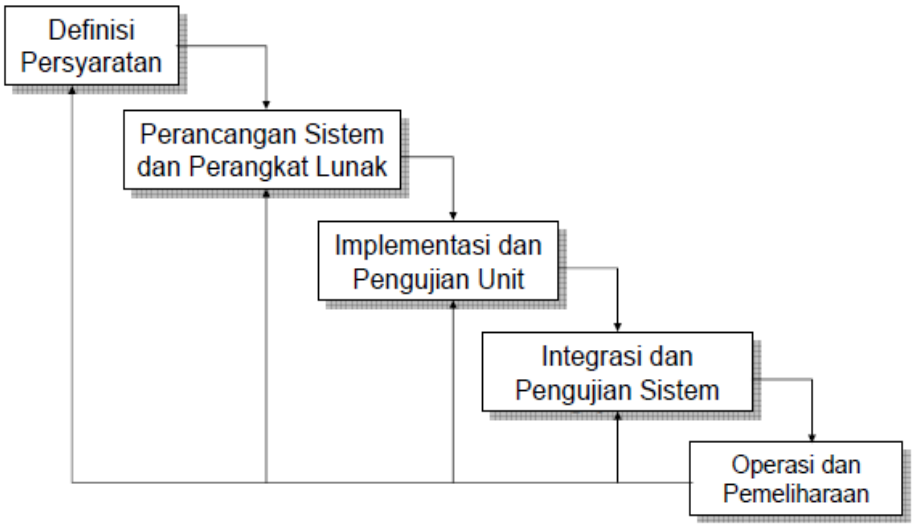
1. Model aliran kerja *(workflow)* → menunjukkan kegiatan pada proses bersama dengan input, output, dan ketergantungannya. Merepresentasikan pekerjaan manusia.

2. Model aliran data *(data flow)* → merepresentasikan proses sebagai suatu set kegiatan yang melakukan transformasi data. Menunjukkan bagaimana input ke proses, misalnya spesifikasi ditransformasi menjadi output, misalnya menjadi desain.

3. Model peran/aksi → merepresentasikan peran orang yang terlibat pada perangkat lunak dan kegiatan yg menjadi tanggung jawab mereka.

Model umum pada proses Perangkat Lunak

1. **Model Sekuensial Linier / air terjun *(waterfall)***



Gambar 2.29 Water fall

Mengambil kegiatan dasar seperti spesifikasi, pengembangan, validasi, evolusi dan merepresentasikannya sebagai fase-fase proses yang berbeda seperti spesifikasi persyaratan, perancangan perangkat lunak, implementasi, pengujian dan seterusnya.

Berisi rangkaian aktivitas proses seperti yang telah diuraikan diatas dan disajikan dalam proses yang terpisah, seperti spesifikasi kebutuhan, implementasi desain perangkat lunak, uji coba dan seterusnya. Setelah setiap langkah didefinisikan, langkah tersebut di *sign off* dan pengembangan dilanjutkan pada langkah berikutnya.

Model ini telah diperoleh dari proses engineering lainnya. Model ini menawarkan cara pembuatan perangkat lunak secara lebih nyata. Langkah-langkah yang penting dalam model ini adalah :

a. Penentuan dan analisis spesifikasi

   Jasa, kendala dan tujuan dihasilkan dari konsultasi dengan pengguna sistem. Kemudian semuanya itu dibuat dalam bentuk yang dapat dimengerti oleh user dan staf pengembang.

b. Desain sistem dan perangkat lunak

   Proses desain sistem membagi kebutuhan-kebutuhan menjadi sistem perangkat lunak atau perangkat keras. Proses tersebut menghasilkan sebuah arsitektur sistem keseluhan.

c. Proses pembuatan program (Coding)

   Setelah melakukan proses desain perangkat lunak, dilanjutkan dengan proses transformasi ke dalam satu atau lebih program dengan menggunakan bahasa pemrograman.

d. Testing dan Implementasi

   Selama tahap ini desain perangkat lunak disadari sebagai sebuah program lengkap atau unit program. Uji unit termasuk pengujian bahwa setiap unit sesuai spesifikasi. Unit program diintegrasikan dan diuji menjadi sistem yang lengkap untuk menyakinkan bahwa persyaratan perangkat lunak telah dipenuhi. Setelah ujicoba, sistem disampaikan ke pelanggan.

2. Model Prototipe

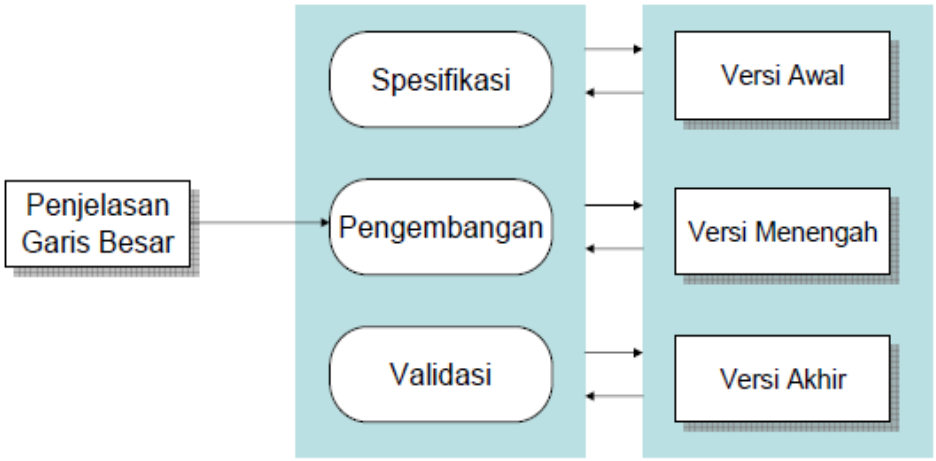
Prototyping paradigma dimulai dengan pengumpulan kebutuhan. Pengembangan dan pelanggan bertemu dan mendefinisikan objek keseluruhan dari perangkat lunak mengidentifikasi segala kebutuhan yang diketahui dan area garis besar dimana definisi lebih jauh merupakan keharusan kemudian dilakukan perancangan kilat.

Secara ideal prototype berfungsi sebagai sebuah mekanisme untuk mengidentifikasi kebutuhan perangkat lunak. Bila prototype yang sedang bekerja dibangun, pengembang harus mempergunakan fragmen-fragmen program yang ada atau mengaplikasikan alat-alat bantu yang memungkinkan program bekerja secara cepat.

3. Model Rapid Application Development (RAD)

Rapid Application Development adalah sebuah model proses perkembangan perangat lunak sekuensial linier yang menekankan siklus perkembangan yang sangat pendek. Model RAD ini merupakan sebuah adaptasi kecepatan tinggi dari model sekuensial linier diamana perkembangan cepat dicapai dengan menggunakan pendekatan konstruksi berbasis komponen.

4. Pengembangan Evolusioner



Gambar 2.30 Pengembangan evolusioner

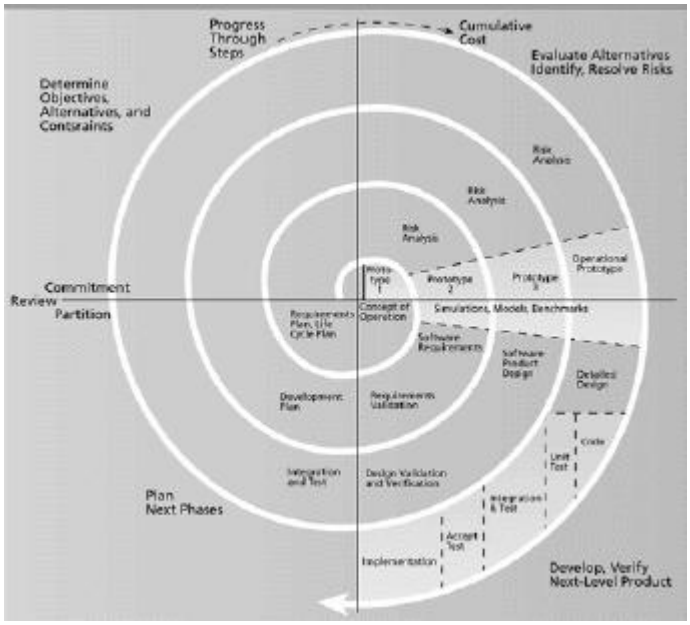
Pendekatan ini berhimpitan dengan kegiatan spesifikasi, pengembangan, dan validasi. Sistem awal dikembangkan dengan cepat dari spesifikasi abstrak. Sistem ini kemudian di perbaiki dengan masukan dari pelanggan untuk menghasilkan sistem yang memuaskan kebutuhan pelanggan.

Pendekatan ini interleaves aktivitas spesifikasi, pengembangan dan validasi. Sistem awal dengan cepat dikembangkan dari pelanggan untuk memproduksi sistem yang memenuhi kebutuhan pelanggan, kemudian sistem disampaikan. Sistem itu mungkin diimplementasikan kembali dengan

pendekatan yang lebih terstruktur untuk menghasilkan sistem yang kuat dan maintable.

Model ini berdasarkan pada ide pengembangan pada implementasi awal yang akan menghasilkan komentar pemakai sehingga dapat dilakukan perbaikan melalui banyak versi sampai sistem yang mencukupi dapat dikembangan. Selain memiliki aktivitas-aktivitas yang terpisah model ini memberikan feedback dengan cepat dan serentak.

5. Model Spiral



Gambar 2.31 Model Spiral

Siklus spiral dimulai dengan penguraian tujuan-tujuan seperti performance, kegunaan, dan seterusnya. Cara alternatif dalam pencapaian tujuan dan hambatan dipergunakan dengan sebaik-baiknya kemudian diperhitungkan. Setiap alternatif diperhitungan bertentangan dengan tujuan. Ini biasanya menghasilkan identifikasi sumber resiko proyek. Langkah selanjutnya adalah mengevaluasi resiko-resiko ini dengan aktivitas seperti analisis yang lebih detail, pembuatan model/contoh, simulasi dan seterusnya.

Untuk menggunakan model spiral, Boehm menyarankan sebuah bentuk umum yang dipenuhi dalam setiap daerah spiral. Bentuk ini mungkin dilengkapi pada sebuah level abtrak atau perkiraan rinci yang imbang dari pengembangan produk.

2.2.6 Bahasa Pemrograman dan Software Pendukung

1. HTML

HyperText Markup Language merupakan standard bahasa yang digunakan untuk menampilkan document web, yang biasa anda lakukan dengan HTML yaitu :

- Mengontrol tampilan dari web page dan contentnya.
- Mempublikasikan document secara online sehingga bisa di akses dari seluruh dunia.
- Membuat online form yang bisa di gunakan untuk menangani pendaftaran, transaksi secara online.
- Menambahkan object-object seperti image, audi, video dan juga java applet dalam document HTML.

Dokumen HTML disebut sebagai markup language karena mengandung tanda-tanda tertentu yang digunakan untuk menentukan tampilan suatu teks tersebut dalam suatu dokumen.

Dokumen HTML disebut sebagai hypertext, ini berarti suatu dokumen tidak harus dibaca secara berurutan dari atas ke bawah. Dokumen dapat dibaca langsung menuju ke topik tertentu secara langsung dalam dokumen dengan menggunakan teks penghubung.

Sejarah HTML

❖ Tahun 1980, IBM memikirkan pembuatan suatu dokumen yang akan mengenali setiap elemen dari dokumen dengan suatu tanda tertentu. IBM kemudian mengembangkan suatu jenis bahasa yang menggabungkan teks dengan perintah-perintah pemformatan dokumen. Bahasa ini dinamakan Markup Langiage, sebuah bahasa yang menggunakan tanda-tanda sebagai basisnya. IBM menamakan sistemnya ini sebagai Generalized Markup Language atau GML.

- Tahun 1986, ISO menyatakan bahwa IBM memiliki suatu konsep tentang dokumen yang sangat baik, dan kemudian mengeluarkan suatu publikasi (ISO 8879) yang menyatakan markup language sebagai standar untuk pembuatan dokumen-dokumen. ISO membuat bahasa ini dari GML milik IBM, tetapi memberinya nama lain, yaitu SGML (Standard Generalized Markup Language).
- ISO dalam publikasinya meyakini bahwa SGML akan sangat berguna untuk pemrosesan informasi teks dan sistem-sistem perkantoran. Tetapi diluar perkiraan ISO, SGML dan terutama subset dari SGML, yaitu HTML juga berguna untuk menjelajahi internet. Khususnya bagi mereka yang menggunakan World Wide Web.

Program editor HTML

Untuk membuat dokumen HTML, anda perlu mengetahui dan mempelajari tag-tag yang digunakan untuk menandai bagian-bagian dari suatu dokumen HTML. Anda dapat menggunakan program editor teks biasa seperti Notepad untuk aplikasi Windows atau Sidekick dan WordStar dalam format Non Document untuk aplikasi DOS.

Semua dokumen HTML harus disimpan sebagai file teks murni. Ada dua macam ekstensi file yang dapat digunakan, yaitu ekstensi file ".html" (untuk sistem operasi UNIX atau WINDOWS 95) atau ".htm" (untuk sistem operasi DOS/WINDOWS 3.x)

2. CSS

Pengenalan CSS

Sebelum Anda dapat menguasai CSS, tentunya Anda harus memahami terlebih dahulu pengertian dari CSS itu sendiri. Anda akan mengetahui sejarah singkat lahirnya CSS, kegunaannya, dan cara penggunaannya.

Sejarah Singkat CSS

Pada tanggal 17 Desember 1996, ***World Wide Web Consortium*** **(W3C)** menjadikan **CSS** sebagai bahasa pemrograman standard dalam pembuatan dokumen web. **Tujuannya** adalah mengurangi pembuatan TAG-TAG baru oleh *Netscape* dan

Internet Explorer (IE), karena pada masa itu kedua *browser* saling bersaing mengembangkan TAG sendiri untuk mengatur tampilan web.

CSS level 1 mendukung pengaturan tampilan dalam hal:

1. Font (jenis, ketebalan),
2. Warna teks, latar belakang, dan elemen lainnya,
3. Text attributes, misalnya spasi antar baris, kata, dan huruf,
4. Posisi text, gambar, tabel, dan elemen lainnya,
5. Marjin, border, dan padding.

Selanjutnya di tahun 1998, W3C menyempurnakan CSS awal dengan menciptakan standard CSS2 (CSS *level 2*) – yang menjadi standard hingga saat ini. Pada CSS *level* 2 ini, di masukkan semua atribut dari CSS1, serta diperluas dengan penekanan pada
International accesibility and capability khususnya *media-specific* CSS. Bahkan pada perkembangannya, saat ini sudah muncul CSS3. Pekerjaan dari CSS3 dimulai pada tahun 2000, tidak lama setelah CSS2 di implementasikan. CSS3 ini sampai sekarang masih terus dikembangkan, spesifikasinya dibagi pada beberapa topik atau modul.

Pengertian & Manfaat CSS

Menurut Wikipedia, Cascading Style Sheets (CSS) adalah: "*bahasa pemrograman untuk mengatur tampilan suatu website atau blog*" Tampilan yang dimaksud disini adalah jenis, ukuran, dan warna *font*; jarak antara baris dan panjangnya, marjin dan indentasi, *background* dan warna, serta masih banyak lagi. **CSS** merupakan fitur yang sangat penting dalam pembuatan Dynamic HTML. Meskipun bukan merupakan keharusan dalam membuat web, namun penggunaan CSS merupakan kelebihan tersendiri. Menggunakan CSS tidak memerlukan perangkat lunak tertentu, karena merupakan *script* yang telah *embedded* dengan HTML. Anda cukup menggunakan aplikasi ***notepad*** untuk menciptakan *script* CSS sendiri.

Manfaat dari CSS:

1. Kode HTML menjadi lebih sederhana dan lebih mudah diatur,
2. Ukuran file menjadi lebih kecil, sehingga *load file* lebih cepat,
3. Mudah untuk merubah tampilan, hanya dengan merubah *file* CSS saja,
4. Dapat berkolaborasi dengan JavaScript dan merupakan pasangan setia XHTML,
5. Digunakan dalam hampir semua *web browser.*

Sintaks & Penempatan Kode CSS

Sintaks CSS terdiri dari tiga bagian:

- *selector*
- *property*
- *value*

Selector adalah elemen atau *tag* HTML yang akan di-definisi-kan. ***Property*** adalah atribut yang akan diganti dengan "nilai" tertentu. Properti dan nilai dipisahkan dengan tanda titik dua (:) dan keduanya diapit oleh tanda kurung kurawal ({). Berdasarkan ketiga bagian tersebut, sintaks dasar CSS dapat dituliskan:

selector { property: value}

contoh 1:

p { color:black }

Jika suatu nilai mengandung **spasi**, nilai tersebut harus diapit dengan tanda petik ("). Untuk memberikan lebih dari satu _property bagi suatu _property, pisahkan setiap kombinasi "property:value" dengan tanda titik koma (;). Untuk membuat definisi gaya lebih mudah dibaca, definisi tiap *property* dapat dilakukan per baris. Sehingga sintaksnya menjadi: selector { property1: value; property2: "value_value"; … }

contoh 2:

p { text-align:center; font-family:"sans serif" }

Aturan-aturan yang berlaku dalam sintaks CSS:

1. Jangan ada spasi antara property value dengan unitnya (contoh 1 & 2),

2. Nama properti bersifat CASE SENSITIVE dan menggunakan huruf kecil,Jika nilai / *value* lebih dari satu kata, maka gunakan tanda titik koma (;) untuk membatasi properti yang satu dengan yang lain (contoh 2),

3. Jika *selector* lebih dari satu dan mempunyai properti dan nilai yang sama, gunakan tanda koma (,).

contoh: h1,h2,h3 {color:green}

Class selector

- Dengan menggunakan *class selector*, Anda dapat mendefinisikan perbedaan style untuk tag elemen html yang sama.
- Class dibuat dengan tanda titik kemudian diikuti dengan nama kelasnya namakelas{ }

contoh 3:

Misalkan Anda ingin membuat dua tipe paragraf dalam dokumen, yang satu **rata kanan**, dan yang lain **rata kiri**. Maka:

p.right {text-align: right}

dan

p.left {text-align: left}

Aturan *Class selector*:

1. Jangan memberi nama class dengan angka,

2. Jika memberikan nama *class* lebih dari satu kata/kalimat, penulisannya menggunakan tanda petik (“).

ID Selector

Aturan penamaan ID:

1. Dapat mengandung huruf, angka, atau karakter garis bawah,
2. Karakter pertama harus berupa huruf atau karakter garis bawah,
3. Diawali dengan tanda #,
4. Jangan memberi nama id sama dengan value,
5. Jangan memberi nama id dengan tag html kemudian diikuti tanda #.

contoh 4:

nama File : id.html

```
<html>
<head>
<style type = "text/css">
<!--
#tebal { font-weight : bold;
color : blue; }
#miring { font-style : italic;
color: pink; }
-->
</style>
</head>
<body>
<h1>Contoh</h1>
<h2 id = "miring">
Selamat Belajar CSS
</h2>
<h3 id = "tebal">
Saya baru pertama kali belajar CSS
</h3>
</body>
</html>
```

Penempatan kode CSS dalam HTML

Ada tiga cara penempatan kode CSS dalam HTML :

1. **Internal CSS**

   Metode penulisan kode CSS langsung dalam *file* HTML.

   contoh:

   nama File :internal.html

   ```
   <html>
   ```

```
<head>
<style type=”text/css”>
p {color: white; }
body {background-color: black; }
</style>
</head>
<body>
<p>Menguak Rahasia CSS</p>
</body>
</html>
```

2. **Eksternal CSS**

Memanggil file CSS dari tempat/folder/lokasi lain, dengan kata lain file CSS terpisah dengan file HTML. Browser akan membaca definisi style dari file test.css dan akan mengikuti format tersebut. Eksternal style ditulis di text editor lain dengan ekstention .css.

contoh:

nama File : eksternal.html

```
<html>
<head>
<link rel=”stylesheet” type=”text/css” href=”test.css”/>
</head>
<body>
<h3>Menguak Rahasia CSS</h3>
</body>
</html>
```

3. **Inline CSS**

Penulisan kode CSS dalam *tag* HTML.

contoh:

nama File : inline.html

```
<html>
```

```
<head>
</head>
<p style=”background: red; color: black;”>Menguak
Rahasia CSS</p>
</body>
</html>
```

Contextual Selectors

• Contextual Selectors digunakan jika style akan digunakan dalam situasi tertentu, misalnya saja jika elemen HTML yang digunakan merupakan sebuah kombinasi. Misalnya saja jika kita akan membuat semua table data mempunyai align center , font arial, dan huruf berwarna orange.

```
<table>
<tr>
        <td></td>
        <td></td>
</tr>
</table>
```

Maka digunakan contextual selectors

Table tr td{ color: orange;
font-family: arial;
Text-align:center;}

Contoh:

Nama File : tabel1.html

```
<html>
<head>
<style type="text/css">
table tr td{color:orange; font-family:arial;text-align:center;}
</style>
</head>
<body>
```

```
<table border="1">
<tr>
<td>Cinta</td>
<td>Bahagia</td>
</tr>
<tr>
<td colspan=2 align=center>Sedih</td>
</tr>
</table>
</body>
</html>
```

3. Java Script

Sejarah JavaScript

JavaScript pertama kali diperkenalkan oleh Netscape pada tahun 1995. Pada awalnya bahasa yang sekarang disebut JavaScript ini dulunya dinamai “LiveScript”” yang berfungsi sebagai bahasa sederhana untuk browser Netscape Navigator 2 yang sangat populer pada saat itu. Kemudian sejalan dengan sedang giatnya kerjasama antara Netscape dan Sun (pengembang bahasa pemrograman “Java”) pada masa itu, maka Netscape memberikan nama “JavaScript” kepada bahasa tersebut pada tanggal 4 desember 1995.

Pada saat yang bersamaan Microsoft sendiri mencoba untuk mengadaptasikan teknologi ini yang mereka sebut sebagai “Jscript” di browser milik mereka yaitu Internet Explorer 3. JavaScript sendiri merupakan modifikasi dari bahasa pemrograman C++ dengan pola penulisan yang lebih sederhana dari bahasa pemrograman C++.

Pengertian JavaScript

JavaScript adalah bahasa pemrograman berbasis prototipe yang berjalan disisi klien. Jika kita berbicara dalam konteks web, sederhananya, kita dapat memahami JavaScript sebagai bahasa pemrograman yang berjalan khusus untuk di browser atau halaman web agar halaman web menjadi lebih hidup. Kalau dilihat dari suku katanya terdiri dari dua suku kata, yaitu Java dan Script. Java adalah Bahasa pemrograman berorientasi objek, sedangkan Script adalah serangkaian instruksi program.

Secara fungsional, JavaScript digunakan untuk menyediakan akses script pada objek yang dibenamkan (embedded). Contoh sederhana dari penggunaan JavaScript adalah membuka halaman pop up, fungsi validasi pada form sebelum data dikirimkan ke server, merubah image kursor ketika melewati objek tertentu, dan lain lain.

Yang Harus Diperhatikan Dalam Penulisan JavaScript

Yang harus diperhatikan dalam pengelolaan pemrograman JavaScript, diantaranya JavaScript adalah “case sensitive”, yang artinya JavaScript membedakan huruf besar dan huruf kecil. Jika Anda pernah belajar bahasa pemrograman seperti Turbo C atau C++, maka sama seperti bahasa pemrograman tersebut, dimana huruf T tidak sama dengan huruf t.

Dalam bahasa pemrograman JavaScript juga, sebagai contoh fungsi perintah var tidak boleh ditulis Var dan juga tidak boleh ditulis VAR (huruf besar semua), yang benar adalah var (huruf kecil semua). Perintah lain adalah new Date tidak boleh ditulis new date (huruf kecil semua), dsb.

Kelebihan JavaScript

JavaScript bekerja pada sisi browser. maksudnya begini : untuk menampilkan halaman web, user menuliskan alamat web di address bar url. setelah itu, browser “mengambil” file html (dengan file jJvaScript yang melekat padanya jika memang ada) ke server yang beralamat di URL yang diketikan oleh user. Selesai file diambil,

file ditampilkan pada browser. Nah, setelah file JavaScript berada pada browser, barulah script JavaScript tersebut bekerja.

Efek dari Javascript yang bekerja pada sisi browser ini, Javascript dapat merespon perintah user dengan cepat, dan membuat halaman web menjadi lebih responsif. JavaScript melakukan apa yang tidak bisa dilakukan oleh HTML, PHP, dan CSS : menangani hal – hal yang membutuhkan respons cepat terhadap aksi dari user.

Contoh : fungsi validasi pada form. ketika anda mengisi sebuah form yang divalidasi menggunakan JavaScript, anda mengetikkan data lalu mengetik submit, sebelum data dikirimkan ke server, data akan "dicek" terlebih dahulu pada browser menggunakan fungsi JavaScript yang ada pada halaman web. sehingga, jika memang data yang anda isikan tidak valid, daripada membuang – buang waktu dengan mengirimkan data ke server baru di validasi di server dan lalu server mengirimkan respons balik mengenai ketidak validan input data anda, lebih baik cek validasi data form dilakukan secara lokal di browser menggunakan fungsi JavaScript.

Lebih Cepat dan Lebih Hemat

Implementasi terpopuler saat ini dari pemrograman JavaScript adalah teknik AJAX. (Asynchronous JavaScript and XMLHTTP). teknik ini sering digunakan oleh aplikasi berbasis web seperti Gmail, Google Reader, dan lain lain. Teknik yang membuat pertukaran data antara server dan browser terjadi di belakang layar sehingga interaksi antara user dan aplikasi web semakin responsif. Post tersendiri untuk membahas hal ini akan kita siapkan.

3. PHP (Personal Home Page)

PHP adalah bahasa pemorgraman yang memungkinkan para web developer untuk membuat aplikasi web yang dinamis dengan cepat. PHP merupakan singkatan dari "PHP: *Hypertext Preprocessor*". PHP ditulis dan diperkenalkan pertama kali sekitar tahun 1994 oleh Rasmus Lerdorf melalui situsnya untuk mengetahui siapa saja yang telah mengakses ringkasan *online*-nya.

PHP merupakan salah satu bahasa script yang terbilang baru dan tersedia secara bebas dan masih memungkinkan untuk dikembangkan lebih lanjut. PHP dapat diintegrasikan *(embedde)* ke dalam web *server*, atau dapat berperan sebagai program CGI yang terpisah. Karakteristik yang paling unggul dan paling kuat dalam PHP adalah lapisan integrasi *database* (*database integration layer*). *Database* yang didukung PHP adalah: *Oracle, Adabas-D, Sybase, FilePro, mSQL, Velocis, MySQL, Informix, Solid, dBase, ODBC, Unix dbm, dan PostgreSQL*.

Tujuan utama penggunaan bahasa ini adalah untuk memungkinkan perancang web menulis halaman web dinamis dengan cepat.

Hubungan PHP dengan HTML

Halaman web biasanya disusun dari kode-kode html yang disimpan dalam sebuah file berekstensi .html. File html ini dikirimkan oleh *server* (atau file) ke *browser*, kemudian *browser* menerjemahkan kode-kode tersebut sehingga menghasilkan suatu tampilan yang indah. Lain halnya dengan program PHP, program ini harus diterjemahkan oleh *web-server* sehingga menghasilkan kode html yang dikirim ke *browser* agar dapat ditampilkan. Program ini dapat berdiri sendiri ataupun disisipkan di antara kode-kode html sehingga dapat langsung ditampilkan bersama dengan kode-kode html tersebut. Program PHP dapat ditambahkan dengan mengapit program tersebut di antara tanda . Tanda-tanda tersebut biasanya disebut tanda untuk *escaping* (kabur) dari kode html. File html yang telah dibubuhi program PHP harus diganti ekstensi-nya menjadi .PHP3 atau .PHP.

PHP merupakan bahasa pemograman web yang bersifat *server-side* HTML = *embedded scripting*, di mana script-nya menyatu dengan HTML dan berada si *server*. Artinya adalah sintaks dan perintah-perintah yang kita berikan akan sepenuhnya dijalankan di *server* tetapi disertakan HTML biasa. PHP dikenal sebgai bahasa scripting yang menyatu dengan tag HTML, dieksekusi di *server* dan digunakan untuk membuat halaman web yang dinamis seperti ASP (*Active Server Pages*) dan JSP (*Java Server Pages*).

Sejarah PHP

PHP pertama kali dibuat oleh Rasmus Lerdroft, seorang *programmer* C. Semula PHP digunakannya untuk menghitung jumlah pengunjung di dalam webnya. Kemudian ia mengeluarkan *Personal Home Page Tools* versi 1.0 secara gratis. Versi ini pertama kali keluar pada tahun 1995. Isinya adalah sekumpulan script PERL yang dibuatnya untuk membuat halaman webnya menjadi dinamis. Kemudian pada tahun 1996 ia mengeluarkan PHP versi 2.0 yang kemampuannya telah dapat mengakses *database* dan dapat terintegrasi dengan HTML. Pada tahun 1998 tepatnya pada tanggal 6 Juni 1998 keluarlah PHP versi 3.0 yang dikeluarkan oleh Rasmus sendiri bersama kelompok pengembang *softwarenya.*

Versi terbaru, yaitu PHP 4.0 keluar pada tanggal 22 Mei 2000 merupakan versi yang lebih lengkap lagi dibandingkan dengan versi sebelumnya. Perubahan yang paling mendasar pada PHP 4.0 adalah terintegrasinya Zend Engine yang dibuat oleh Zend Suraski dan Andi Gutmans yang merupakan penyempurnaan dari PHP *scripting engine*. Yang lainnya adalah *build in* HTTP *session*, tidak lagi menggunakan *library* tambahan seperti pada PHP. Tujuan dari bahasa *scripting* ini adalah untuk membuat aplikasi-aplikasi yang dijalankan di atas teknologi web. Dalam hal ini, aplikasi pada umumnya akan memberikan hasil pada web *browser*, tetapi prosesnya secara keseluruhan dijalankan web *server*.

Kelebihan PHP

Ketika *e-commerce* semakin berkembang, situs-situs yang statispun semakin ditinggalkan, karena dianggap sudah tidak memenuhi keinginan pasar, padahal situs tersebut harus tetap dinamis. Pada saat ini bahasa PERL dan CGI sudah jauh ketinggalan jaman sehingga sebagian besar designer web banyak beralih ke bahasa *server-side scripting* yang lebih dinamis seperti PHP. Seluruh aplikasi berbasis web dapat dibuat dengan PHP. Namun kekuatan yang paling utama dari PHP adalah pada konektivitasnya dengan sistem *database* di dalam web.

Sistem *database* yang dapat didukung oleh PHP adalah :

1. *Oracle*
2. *SQL*
3. *MySQL*
4. *PostgreSQL*
5. dan lainnya

Script PHP

Setiap program PHP disebut dengan script. Script berupa file text yang dapat dibuat dengan menggunakan program editor file text biasa seperti notepad, edit, dan lainya. Script PHP diawali dengan tag <? Dan diakhiri dengan tag ?>. Setiap baris atau statement harus diakhiri dengan menggunakan tanda titik koma (;) dan umumnya setiap statement dituliskan dalam satu baris. Script PHP merupakan script yang digunakan untuk menghasilkan halaman-halaman web. Cara penulisan script juga dibedakan menjadi 2, yaitu *Embedded Script* dan *Non Embedded Script*.

1. *Embedded Script*

   Adalah script PHP yang disisipkan diantara tag-tag HTML. Script PHP digunakan apabila isi dari suatu dokumen HTML diinginkan dari hasil eksekusi suatu script PHP

```
<html>
<head>
<title>contoh</title>
</head>
<body>
<?php
echo “Website Mandiri Motor”;
?>
```

```
</body>
</html>
```

2. *Non Embedded Script*

   Script PHP dalam *non embedded script* ini digunakan sebagai murni pembuatan program dengan PHP, tag HTML yang dihasilkan untuk membuat dokumen merupakan bagian dari script PHP.

Tipe Data PHP

PHP mengenal lima tipe data, yaitu : *integer, floating point, string, arrays* dan *Objects*. Tipe data dari sebuah variabel akan ditentukan otomatis oleh PHP bergantung pada operasi yang sedang dilakukan mengunakan variable tersebut.

1. Tipe Data *Integer*

   Tipe data *integer* meliputi semua bilangan bulat, range data *integer* pada PHP yaitu antara –2,147,483,648 samapai +2,147,483,647 pada platform 32 bit. PHP akan secara otomatis mengkonversi data *integer* menjadi *floating point* jika berada diluar range diatas.

2. Tipe Data *Floating Point*

   Floating point merepresentasikan bilangan pecahan atau bilangan desimal, range data *floating point* pada PHP yaitu antara 1.7E-308 sampai 1.7E+308.

3. Tipe Data *String*

   Sebuah data dengan tipe *string* dinyatakan dengan mengapitnya mengunakan tanda petik tunggal (‘ ‘) maupun ganda (“ “). Perbedaannya adalah, jika kita tanda petik tunggal, maka pada *string* itu tidak dapat dimasukkan variabel dan escape sequence handling.

4. Tipe Data *Array*

   Array adalah sebuah data yang mengandung satu atau lebih data, dan dapat diindeks berdasarkan numerik maupun *string*. Data yang dikandung pada sebuah *array* dapat dari tipe data yang lain bahkan dapat juga bertipe *array*.

PHP membolehkan elemen dari *array* terdiri atas data dengan tipe yang berbeda-beda.

5. Tipe Data *Object*

   Object adalah sebuah tipe data yang dapat berupa sebuah bilangan, variabel atau bahkan sebuah fungsi. *Object* dibuat dengan tujuan untuk membantu programmer yang terbiasa dengan *Object Oriented Programming*.

Struktur Kendali PHP

Struktur Kendali atau statement adalah bagian penting dalam suatu bahasa pemrograman, karena bagian ini mengatur jalurnya eksekusi suatu program. Dalam bahasa pemrograman PHP dikenal 4 jenis pernyataan utama, yaitu if, switch, while, dan for.

WAMP

Kepanjangan dari *WAMP* yaitu *Windows* + *Apache* + *MySQL* + *PHP*. *WAMP* merupakan tool yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dengan menginstall *WAMP* maka tidak perlu lagi melakukan instalasi dan konfigurasi web *server APACHE*, PHP dan *MySQL* secara manual. *WAMP* akan menginstalasi dan mengkonfigurasikannya secara otomatis untuk anda atau auto konfigurasi.

APACHE

APACHE merupakan salah satu jenis program yang bertujuan untuk mengirimkan kembali sintaks yang telah diberikan oleh user dan menampilkan kembali dalam bentuk dalam format hasilnya saja. Program *APACHE* ini hanya terdapat pada sebuah *server* yang terdapat pada ISP (Internet Service Provider) atau di Indonesia disebut Jasa Penyelenggara Internet. Saat ini program yang paling banyak terdapat pada *server-server* di dunia adalah *APACHE*. Hal ini disebabkan karena beberapa keunggulan yang diberikan.

Keunggulan-keunggulan tersebut antara lain :

1. Memiliki kemapuan pembacaan yang tinggi dan mudah disesuaikan

2. Memiliki tingkat keamanan yang tinggi. Hal ini dibutuhkan bagi para penyimpan data yang memiliki tingkat kerahasian tertentu.
3. Dapat dijalankan untuk berbagai macam operasi sistem.

APACHE merupakan suatu nama yang diambil dari istilah "*A PAtCHy*" *server* yang dapat diartikan dengan mengambil suatu patch atau bagian dari program yang diberikan.

APACHE sudah berkembang sejak versi pertamanya. Sampai saat ditulisnya artikel ini versi terakhirnya yang ada yaitu *APACHE* ver 2.0.54. *APACHE* bersifat open source, artinya setiap orang boleh menggunakannya, mengambil dan bahkan mengubah kode programnya.

Tugas utama *APACHE* adalah menghasilkan halaman web yang benar kepada peminta, berdasarkan kode PHP yang dituliskan oleh pembuat halaman web. Jika diperlukan juga berdasarkan kode PHP yang dituliskan, maka dapat saja suatu *database* diakses terlebih dahulu (misalnya dalam *MySQL*) untuk mendukung halaman web yang dihasilkan.

Perkembangannya disebut *SQL* yang merupakan kepanjangan dari *Structured Query Language*. *SQL* merupakan bahasa terstruktur yang khusus digunakan untuk mengolah *database*. *SQL* pertama kali didefinisikan oleh *American National Standards Institute* (ANSI) pada tahun 1986. *MySQL* adalah sebuah sistem manajemen *database* yang bersifat open source. *MySQL* adalah pasangan serasi dari PHP. *MySQL* dibuat dan dikembangkan oleh *MySQL* AB yang berada di Swedia.

MySQL dapat digunakan untuk mengelola *database* mulai dari yang kecil sampai dengan yang sangat besar. *MySQL* juga dapat menjalankan perintah-perintah *Structured Query Language* (*SQL*) untuk mengelola *database-database* yang ada di dalamnya. Hingga kini, *MySQL* sudah berkembang hingga versi 5. *MySQL* 5 sudah mendukung *trigger* untuk memudahkan pengelolaan tabel dalam *database*.

Sebuah Website yang interaktif dan dinamis, tentunya akan membutuhkan penyimpanan data yang fleksibel dan cepat untuk diakses. Salah satu *database* untuk *server* adalah *MySQL*. *MySQL* adalah merupakan aplikasi RDBMS (*Relational Data Base Management System*) yang sangat cepat dan dapat diandalkan dalam menangani basis data. Jenis *database* ini sangat popular dan digunakan pada banyak website .

MySQL menggunakan bahasa *SQL* (*Struktur Query Language*) dan bersifat *free* (gratis atau tidak perlu membayar untuk menggunakannya). Selain itu, *MySQL* dapat berjalan di berbagai platform, antara lain Linux, Windows, dan lain sebagainya.

Melakukan koneksi kepada *database MYSQL* dengan menggunakan PHP menggunakan dua tahap :

1. Koneksi kepada *server MySQL*

   Dilakukan untuk melakukan otentikasi, digunakan untuk login ke dalam *server*, menggunakan *MySQL*_connect( ).

2. Koneksi kepada *database*

Dilakukan untuk memilih *database* mana yang ada dalam *server* yang akan dimanipulasi, karena bisa saja *database* yang dimaksud sudah tidak ada. Fungsi yang digunakan adalah *MySQL*_select_db( ) .

Keistimewaan *MySQL*

Sebagai *server database* dengan konsep *database* modern, *MySQL* memiliki keistimewaan. Beberapa keistimewaan dimiliki *MySQL* sebagai berikut:

1. *Portability*

   Database MySQL berfungsi dengan stabil tanpa kendala, berarti berlaku pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac OS X *Server*, Solaris, Amiga, HP-Unix, dan lain-lain

2. *Open Source*

   MySQL merupakan *database* open source (gratis), di bawah lisensi GPL sehingga dapat memperoleh dan menggunakannya secara cuma-cuma tanpa membayar sepersen pun.

3. *Multiuser*

   MySQL merupakan *database* yang dapat digunakan untuk menangani beberapa user dalam waktu bersamaan tanpa mengalami masalah. Dan memungkinkan sebuah *database server MySQL* dapat diakses client secara bersamaan pula.

4. *Performance Tuning*

   MySQL mempunyai kecepatan yang cukup baik dalam menangani query-query sederhana, serta mampu memproses lebih banyak *SQL* per satuan waktu.

5. *Column Type*

   Database MySQL didukung dengan tipe data yang sangat kompleks, seperti signed/unsigned *integer*, float, double, char, varchar, text, blob, data, time, datetime, timestamp, year, set serta enum

6. *Command And Functions*

   MySQL server memiliki operator dan fungsi secara penuh yang mendukung perintah SELECT dan WHERE dalam query.

7. *Security*

   Sistem *Security* pada *MySQL* mempunyai beberapa lapisan sekuritas seperti tingkatan *subnetmask, hostname*, dan izin akses user dengan sistem perizinan yang mendetil serta *password* terenkripsi.

8. *Scalability dan Limits*

   MySQL mempunyai kemampuan menangani *database* dalam skala cukup besar, dengan jumlah record lebih dari 50 juta dan 60 ribu tabel serta 5 miliar baris. Selain itu dapat menampung indeks sampai 32 indeks pada tiap tabelnya.

9. *Connectivity*

   Adanya kemampuan *MySQL* melakukan koneksi dengan *client* menggunakan protokol TCP/IP, Unix socket (Unix), atau Named Pipes (NT).

10. *Localization*

    Adanya kemampuan dalam mendeteksi kesalahan (*error code*) pada *client* menggunakan lebih dari dua puluh bahasa.

11. *Interface*

    MySQL memiliki *interface* terhadap berbagai aplikasi dan bahasa pemograman menggunakan fungsi API (*Application Programming Interface*).

12. *Clients dan Tools*

    Database MySQL dilengkapi berbagai tools yang dapat digunakan untuk administrasi *database*.

13. *Struktur Tabel*

    MySQL memiliki struktur tabel yang lebih fleksibel dalam menangani ALTER TABLE dibandingkan *database* lainnya.

PHPMyAdmin

Pengelolaan *database* dengan *MYSQL* harus dilakukan dengan mengetikkan baris-baris perintah yang sesuai (*command line*) untuk setiap maksud tertentu. Jika ingin membuat *database*, pengguna harus mengetikkan baris perintah yang sesuai untuk membuat *database*. Tetapi hal tersebut tentu cukup menyulitkan karena pengguna harus hafal dan mengetikkan perintahnya satu persatu.

Banyak sekali perangkat lunak yang dapat dimanfaatkan untuk mengelola *database* dalam *MySQL*, salah satunya adalah PHP*MyAdmin*. Dengan PHP*MyAdmin* pengguna dapat membuat tabel, mengisi data dan lain-lain dengan mudah tanpa harus hafal perintahnya. Untuk mengaktifkan PHP*MyAdmin* langkah-langkahnya adalah : setelah meng-*install* WAMP dijalankan di *browser* yang ada *(IE, Mozilla Firefox* atau *Opera)* lalu ketikkan alamat web berikut : http://localhost/PHPmyadmin/ pada address bar lalu tekan Enter. Langkah ketiga apabila telah nampak *interface* (tampilan antar muka) PHP*MyAdmin* pengguna bisa memulainya dengan mengetikkan nama *database*, nama tabel dan seterusnya.

Macromedia Dreamweaver 8

Macromedia Dreamweaver merupakan salah satu software dari kelompok Macromedia yang banyak digunakan untuk mendesain situs Web. Adapun *Macromedia Dreamweaver* itu sendiri adalah sebuah HTML editor professional untuk mendesain secara visual dan mengelolah situs atau halaman Web. *Dreamweaver 8* memiliki performa yang lebih baik dan memiliki tampilan yang memudahkan anda untuk membuat halaman web, baik dalam jendela desain maupun dalam jendela kode rumus. *Dreamweaver 8* didukung dengan cara pemakaian yang praktis dan standar, dan juga didukung untuk pengembangan penggunaan CSS, XML, dan RSS, dan kemudahan-kemudahan lain yang diperlukan.

Dreamweaver merupakan *software* yang digunakan oleh Web desainer maupun Web programmer dalam mengembangkan Web. Hal ini disebabkan ruang kerja, fasilitas, dan kemampuan *Dreamweaver* yang mampu meningkatkan produktivitas dan efektivitas dalam desain maupun membangun sebuah situs Web.